iSBC 501™ DIRECT MEMORY ACCESS CONTROLLER HARDWARE REFERENCE MANUAL

Manual Order Number: 9800294-02

Copyright © 1976, 1979 Intel Corporation

Intel Corporation, 3065 Bowers Avenue, Santa Clara, California 95050

Additional copies of this manual or other Intel literature may be obtained from:

Literature Department
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

The information in this document is subject to change without notice.

Intel Corporation makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Intel Corporation assumes no responsibility for any errors that may appear in this document. Intel Corporation makes no commitment to update nor to keep current the information contained in this document.

Intel Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in an Intel product. No other circuit patent licenses are implied.

Intel software products are copyrighted by and shall remain the property of Intel Corporation. Use, duplication or disclosure is subject to restrictions stated in Intel's software license, or as defined in ASPR 7-104.9(a)(9).

No part of this document may be copied or reproduced in any form or by any means without the prior written consent of Intel Corporation.

The following are trademarks of Intel Corporation and may be used only to describe Intel products:

i
ICE
iCS
Insite
Intel
Intelevision
Intellec
iSBC
Library Manager
MCS
Megachassis
Micromap
Multibus
Multimodule
PROMPT
Promware
RMX
UPI
μScope

and the combination of ICE, iCS, iSBC, MCS, or RMX and a numerical suffix.

A128/1079/2K DD

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

The DMA Board provides a direct memory access capability for the high-speed transfer of 8 or 16-bit data. Once a DMA operation is initiated by the central processor unit (CPU), the DMA Board controls the actual transfer of up to 65,536 words of data between memory and an external device, without any further intervention of the CPU required. The DMA Board can "steal" cycles by requesting control of the system bus for each word transferred; or the CPU can, prior to the beginning of a transfer operation, invoke an override capability for the DMA Board. In this case, the DMA Board retains control of the bus until the entire block of data is transferred. After the entire transfer is completed, the CPU would, in response to a DMA interrupt, reset the override capability. This mode of operation allows for "burst" mode transfers to/from very high-speed peripherals.

While the data paths between the DMA Board and the external devices are only 8 bits wide, the bidirectional data bus between the board and the CPU can be either 8 or 16 bits wide. When transferring 16-bit data from memory to a device, the DMA Board disassembles the word and transfers the two bytes separately. Conversely, the board assembles two consecutive bytes from a device and sends the 16-bit word in parallel to memory.[1]

The DMA Board includes provisions that allow it to be interrupt driven. In fact, the DMA interrupt request can be asserted on any one of eight interrupt priority levels.[2] A DMA interrupt request can originate in the external device, in the DMA Board itself (upon completion of a transfer operation), or can be generated by the program being executed in the CPU. The CPU program can enable/disable interrupts or reset an existing interrupt request. The external device can also disable the presentation of interrupts to the CPU by the DMA Board.

In addition to providing a high-speed data path between memory and peripheral devices, the DMA Board includes five I/O ports that allow the CPU to directly address and access five devices (or groups of devices). The fifth port is associated with a 4-bit tag register. When this fifth I/O port is addressed, the contents of the tag register can be used to "steer" the input or output strobe to one of 16 additional devices; thus expanding the I/O capability of the DMA Board. These I/O ports between the CPU and the peripheral devices are usually used to initialize or test the device for a transfer operation, by sending control or address information (e.g., the sector address for a disk) to the device or by reading a status word from the device.

While the DMA Board has been designed specifically as an option for use in SBC 80 and INTELLEC® Microcomputer Development Systems, the board is flexible and powerful enough to be used in many different 8 or 16-bit computer systems that require a direct memory access capability. Consequently, the DMA Board, like all other SBC Boards, is independently available on an OEM basis.

The board is implemented on a single 12-in. X 6.75-in. printed circuit board. The board requires only +5 VDC power.

1.0 FUNCTIONAL ORGANIZATION OF THE DMA BOARD

The DMA Board can be viewed, for descriptive purposes, as consisting of 11 functional blocks:

(1) Address decoding logic
(2) XACK/ generation logic
(3) Control register
(4) Length register
(5) Memory address register
(6) Tag register
(7) Status logic
(8) Bus in/out logic
(9) DMA interrupt logic
(10) DMA transfer control logic
(11) Bus interface logic

[1] For use in Intel SBC or INTELLEC® Systems, an 8-bit word would be used. 16-bit capability is included on the SBC 501 DMA Controller to allow stand-alone use in other OEM systems.

[2] For use with the single interrupt level of the SBC 80/10 Single Board Computer, interrupt level 1 is used.

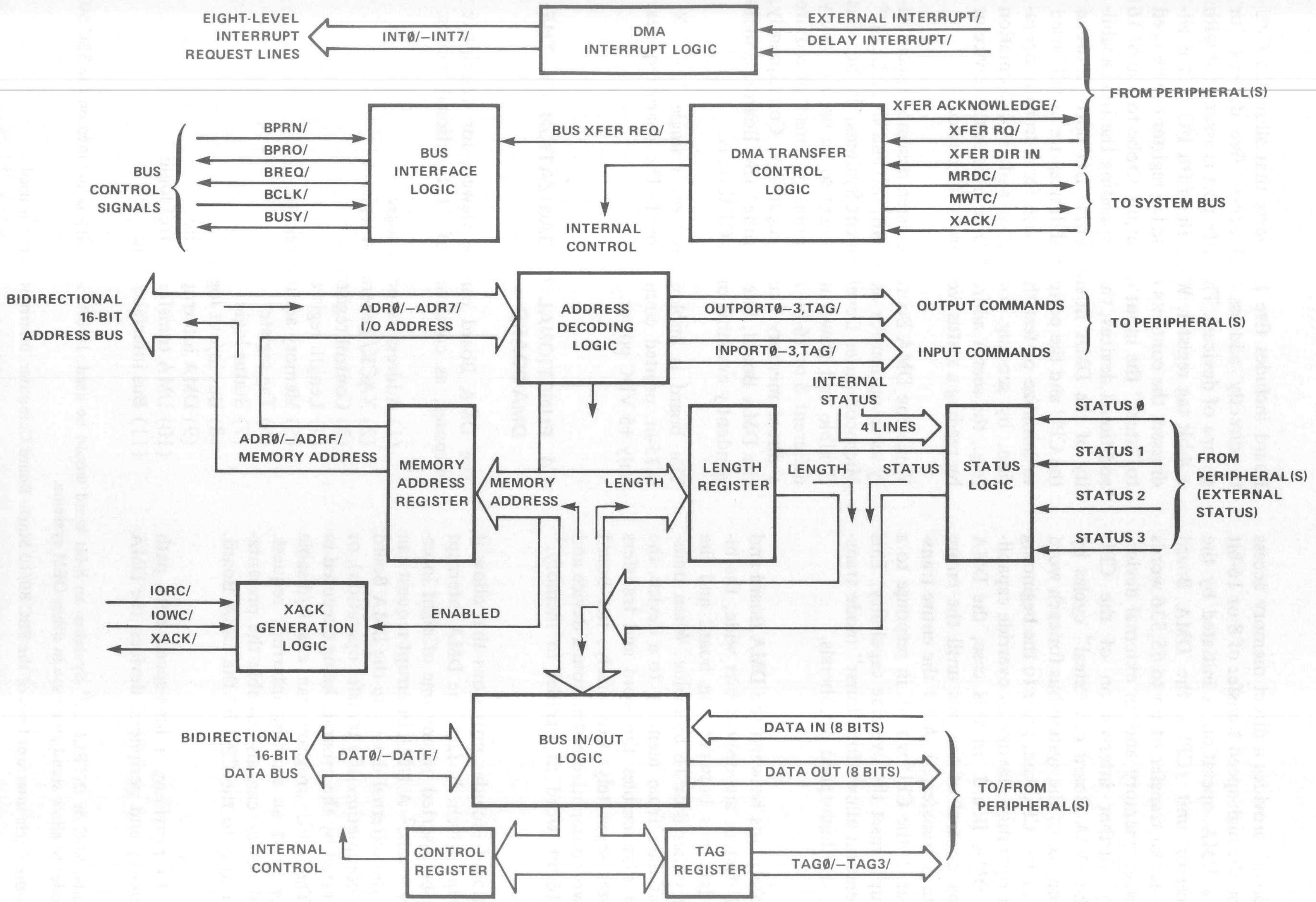


Figure 1. DMA Board Functional Block Diagram

as illustrated in Figure 1.

Before beginning a DMA transfer operation, the DMA Board must be initialized by the CPU. Only after the proper parameters have been loaded into the board can the actual block transfer begin. Once initialized by the CPU, however, the DMA Board can transfer up to 65,536 words of data (8 or 16-bit) between memory and an external peripheral device without any further intervention required of the CPU.

The execution of a direct memory access (DMA) operation requires considerable interaction between the 11 functional blocks that we listed. Consequently, before defining the specific responsibilities of each functional block, we will step through the general sequence of events in a DMA operation.

Sequence of Events

A typical DMA operation is initiated and executed as follows:

(1) The CPU reads the status of the DMA Board and the peripheral device to determine that both are available for use. The CPU reads status by executing an I/O read (input) instruction addressed to port "BASE+6", where BASE is the jumper-selectable base address of the DMA Board. (In a multi-processor system, the CPU should use its bus override feature during the Busy Test/Load Length Register/Busy Set sequence to prevent detrimental interaction with the DMA Board by other CPUs.)

(2) The CPU outputs two 8-bit bytes, specifying the number of words to be transferred, into the 16-bit length register. The least significant byte is loaded (I/O output instruction) to port "BASE+C$_{16}$". The most significant byte is loaded by executing an output instruction to port "BASE+D$_{16}$".

CAUTION

Length register must be loaded *before* busy bit in the control word is set to avoid an inadvertent interrupt.

(3) The CPU loads a control word into the DMA Board's control register. This sets the busy indicator and specifies the type of operation to be performed. The control register is accessed by executing an I/O write (output) instruction addressed to port "BASE+A$_{16}$".

(4) The CPU outputs a 16-bit memory address, specifying the first memory location to be accessed. The least significant byte is loaded into the memory address register by executing an I/O write (output) instruction to port "BASE+E$_{16}$". The most significant byte is loaded by executing an output instruction to port "BASE+F$_{16}$".

(5) The CPU must also provide the peripheral device with any parameters it may need for the operation (e.g., the sector number for a disk). When the CPU executes an output instruction to address "BASE+port#" (where port# equals 0,1,2,3 or B$_{16}$*), the DMA Board routes the byte from the CPU to the peripheral device and generates an output pulse that strobes the byte into the device.

(6) The last step in the initialization portion of the operation is the issuance of a "go" command to the peripheral device. Normally, the CPU issues a "go" command by outputting a command byte to the appropriate port as described in Step 5, but in some configurations a bit in the tag register could be used to notify the device. The CPU can load the tag register by executing output instructions to port "BASE+B$_{16}$".

(7) Each time that the external device requires transfer of a data word, it issues a transfer request (XFER RQ/) to the DMA Board's DMA transfer control logic. The DMA Board's bus interface logic, in turn, requests use of the system bus. After the DMA Board gains control of the system bus, the DMA transfer control logic issues a memory read (MRDC/) or write (MWTC/) command. After the data word is transferred to/from the peripheral device (via the DMA Board's bus in/out logic), the DMA transfer control logic issues a trasnfer acknowledge signal (XFER ACKNOWLEDGE) to the peripheral device. The DMA Board decrements the word count in its length register and increments the value in its memory address register. This sequence continues until the word count equals zero.

*BASE+B$_{16}$ = address of tag register.

(8) When the word count in the DMA Board's length register is decremented to zero, the interrupt latch in the DMA Board is set. If the peripheral is asserting its delay interrupt signal (DELAY INT/), the interrupt request is not passed on to the CPU until the delay signal goes false. If the enable interrupt bit in the DMA Board's control register is set, the interrupt request can be asserted on any one of the eight interrupt levels (the level is jumper selected). If DMA interrupts are not enabled, the CPU must periodically interrogate DMA status to determine when the transfer has been completed.

(9) When the CPU determines that the transfer is complete, it CALLS a service routine. At the end of the service routine, the CPU issues a reset interrupt command to the DMA Board by executing an output instruction to port "BASE+9". The reset interrupt command resets the DMA Board's busy latch and the interrupt latch. The reset interrupt command will also reset the set interrupt latch if the CPU had initiated an interrupt request in the DMA Board. The DMA Board is now ready for the next operation.

Functional Block Descriptions

As we saw in the above sequence of events, the CPU accesses the various blocks within the DMA Board by executing I/O instructions directed to a jumper-selectable block of 16 dedicated port addresses. The *address decoding logic* receives the port address output by the CPU on the bidirectional address bus and generates the appropriate internal control signal for the DMA Board, or generates the proper input/output strobe for the peripheral device. The CPU only accesses the DMA Board to initialize it prior to a transfer operation or to reset it after a transfer operation has been completed. The DMA Board, itself, controls the actual transfer of data between memory and an external peripheral device.

Whenever the CPU accesses an I/O port, it waits for the port to return a transfer acknowledge signal (XACK/) before completing the execution of the I/O instruction. The *XACK/ generation logic*, as its name implies, is responsible for returning the XACK/ signal, whenever the CPU executes an I/O instruction to a port address within the range defined by the DMA Board base address. The actual timing for the generation of XACK/ is jumper-selectable. In addition to generating XACK/ in response to I/O instructions from the CPU, the XACK/ generation logic accepts XACK/ signals generated by memory during DMA data transfers and passes this acknowledgement on to the DMA transfer control logic.

The *control register* is loaded by the CPU, prior to the beginning of a DMA operation. The contents of the control register specify the busy status of the DMA Board, the type of operation to be performed (transfer or non-transfer), the transfer direction (to or from memory), the word size (8 or 16 bits), the interrupt condition (enabled or disabled), and the means by which the DMA Board is using the system bus (contention basis or override basis).

The 16-bit *length register* is loaded by the CPU, prior to a transfer operation, with the word count value specifying the total number of words to be transferred. The word count is decremented by one after each word is transferred. The transfer stops when the word count equals zero.

The 16-bit *memory address register* is loaded by the CPU, prior to a transfer operation, with the address of the first memory location to be accessed. The address is incremented by one for each word transferred. The current 16-bit address is gated onto the bidirectional system address bus and sent to memory during each transfer sequence.

The *tag register* is a general-purpose 4-bit register. The contents of the tag register can be made available to all of the external peripheral devices being controlled by the DMA Board. The tag register can be used in a number of ways. The four tag lines can be used as control lines to the devices (e.g., as the "go" command line to each of four devices). The tag register might be used to expand the maximum number of DMA peripherals by 16. In this case, the tag register would store the select code for one of 16 additional devices. Only that device which recognized its 4-bit select code on the tag lines would respond to command bytes output on the data out bus with the OUTPUT TAG/ strobe.

The *status logic* groups together four internal status bits from the DMA Board (SET INT/, MEM WRT/, INTERRUPT STATUS/, and DMA BUSY/) and four status bits from the external peripheral device, and multiplexes these 8 bits onto the system data bus when directed to do so by the CPU (i.e., when the CPU executes the IN PORT instruction, where PORT = BASE+6).

The *bus in/out logic* routes all data flow to, from or through the DMA Board to its intended destination. Control or address information that is output on the system data bus by the CPU is buffered in the bus in/out logic and directed to registers within the DMA Board (e.g., the length register), or routed out to the peripheral devices via the data out bus (e.g., a "go" command word). Data output by memory during a DMA transfer sequence is received from the bidirectional system data bus, buffered, then routed onto the data out bus and out to an external device. Data input by a peripheral device during a DMA transfer sequence is received from the data in bus, buffered, then routed onto the bidirectional system data bus which carries the data to memory. Status, address or word count information from the DMA Board or from an external device is directed through the bus in/out logic and onto the system data bus which carries it to the CPU.

The *DMA interrupt logic* accepts interrupt requests generated by the CPU program, an external device, or by the DMA Board at the end of a DMA transfer operation (i.e., when the length register contents equal zero). The DMA interrupt logic will only respond (i.e., set the interrupt latch) to an interrupt request originating in the CPU program if the DMA Board is not busy. If the contents of the DMA Board's control register indicate that interrupts are enabled, an interrupt request (regardless of its source) can be asserted on any one of the eight interrupt levels (INTØ/—INT7/). An interrupt level is selected by installing a jumper in the DMA interrupt logic. If the DELAY INT/ line from the external devices is true however, the interrupt request to the CPU will be delayed until DELAY INT/ goes false.

The *DMA transfer control logic* is responsible for the "handshaking" exchanges with the peripheral device and memory during all DMA operations. The DMA transfer control logic accepts transfer request (XFER RQ/) and transfer direction (XFER DIR IN) signals from the external device, generates the read (MRDC/) or write (MWTC/) command for memory, and uses the memory's acknowledgement (XACK/) to generate a transfer acknowledge signal (XFER ACKNOWLEDGE/) for the external device. The DMA transfer control logic also notifies the bus interface logic when it needs the system bus for a data transfer (i.e., BUS XFER RQ/ goes true).

The *bus interface logic* allows the DMA Board to operate as a bus master; that is, to obtain control of the system bus at the exclusion of all other master boards, including the primary CPU. When the DMA Board requires the bus for a DMA data transfer (i.e., when BUS XFER RQ/ is true), the bus interface logic issues the bus request signal (BREQ/). When the bus priority in signal (BPRN/) indicates that no higher priority board is requesting the bus, the bus interface logic sets its bus busy latch (BUSY/) and informs the other logic on the DMA Board that it has been selected. The DMA Board now has control of the bus for one data word transfer sequence. If the DMA Board is to require uninterrupted use of the system bus (e.g., burst mode transfers), an override capability can be extended to the board by the CPU. Override is enabled by setting a bit in the DMA Board's control register. Operation of the bus interface logic is referred to the bus clock signal (BCLK/), which must be supplied by another board in the system.

2.0 DMA BOARD: THEORY OF OPERATION

The following sub-sections provide a complete description of the theory of operation for each of the functional units on the DMA Board.

The DMA Board accepts/transmits signals, data and power through three different PC edge connectors:

- J1 Peripheral connector (to/from I/O peripherals)
- P1 Bus connector (to/from the system bus)
- P2 Auxiliary connector (to/from the auxiliary bus)

To avoid any ambiguity when referring to connector pins in subsequent paragraphs, we will always list the connector as well as the pin whenever such references are required; for example, P1-14 refers to pin 14 on connector P1. Pin lists for the three connectors are provided in Section 3.2.

The schematic (3 sheets) for the DMA Board is provided in Figure 9, located in Section 2.12.

2.1 ADDRESS DECODING LOGIC

The address decoding logic consists of four 3205 three-to-eight decoders, a user installed address jumper, a jumper plug, and assorted gating and buffer circuits, as shown on sheet 2 of the board schematic, Figure 9.

The CPU sends control or address information to, or receives status information from, the DMA Board or one of the attached peripheral devices by executing an I/O instruction directed to a dedicated 8-bit port address. The address decoding logic examines address lines ADRØ/–ADR7/ (pins P1-51 through P1-58) and generates the appropriate control or strobe signal.

Address line ADR7/ is applied to one pole of the X2 jumper plug in an active-low state, and is also inverted and applied to another pole of X2 in an active-high state. Address lines ADR4/, ADR5/ and ADR6/ are applied to the address inputs of one 3205 decoder. The three address inputs cause a low level to appear on one of the eight decoder outputs. Each of these outputs are tied to a jumper wire post; position 9 is considered off. The outputs from the X1 jumper and the X2 jumper plug feed a 7427 negative-input AND gate. The output of the 7427 section (A64-12), in conjunction with address line ADR3/ and the I/O write (IOWC/) or I/O read (IORC/) command, enables the other three 3205 decoders in the address decoding logic. The range of port address that will actually activate the 7427 gate is dependent on the base address for the DMA Board (address bits 4—7) as determined by the position of the X1 and X2 jumpers. Table 1 correlates all of the possible jumper positioning combinations with the base address that they define.

The three least significant address lines (ADRØ/–ADR2/) are applied to the address inputs on three other 3205 decoders. As we mentioned above, address line ADR3/ (active-low) or its complement, ADR3 (active-high), feeds one of the enable inputs on the three decoders. The I/O write command (IOWC/) feeds the other enable input on the first two decoders (output ports), while the I/O read command (IORC/) feeds an enable input on the third decoder (input ports).

The first decoder produces the five output strobes, OUTPORTØ/–OUTPORT3/ and OUTPORT TAG/. The five most significant decoder outputs are buffered, inverted and gated with OUTP PULSE to produce the OUTPORT strobes. OUTP PULSE is generated in the XACK/ generation logic. Timing for the OUTPORT strobes is dependent on the setting of the time base jumper pad in the XACK/ generation logic; that is, an OUTPORT strobe will occur 100, 200, 400, 800 or 1600 ns after the I/O write command (IOWC/) is received, and will remain true for 200, 400, 800, 1600, or 3200 ns, respectively, according to the time base setting (see Section 2.2, Figure 2).

The eight outputs from the second decoder constitute eight internal control signals for the DMA Board, as listed in Table 2.

The eight outputs from the third decoder form the five input strobes to the external devices, INPORTØ/–INPORT3/ and INPORT TAG/, as well as the read DMA status (RD DMAC STAT/) and read length register (RD LEN MSBY/ and RD LEN LSBY/) internal command signals. RD LEN MSBY/ and RD LEN LSBY/ (read most and least significant bytes) are NORed together to form INPUT EN1/. INPUT EN1/, in turn, is NANDed with IORC, ADR3/, and the enable signal (ENABLED) from the 7427 base address gate (A64-12) to form the INPUT EN2/ signal. INPUT EN1/ allows the contents of the length register to be read, and INPUT EN2/ allows the DMA status word to be read (INPUT EN1/ and INPUT EN2/ enable multiplexers shown on sheet 3 of the board schematic).

Table 2 lists the internal control signals and external strobes, with the dedicated port addresses that select each one.

Table 1

DMA BOARD BASE ADDRESS SELECTION

BASE ADDRESS (HEX)	PORT ADDRESS RANGE (HEX)	X1 (JUMPER S20 to S24 THROUGH S2-8)	X2 (JUMPER 28-29-30)
00	00–0F	S2-0 to S2-1	29–30
10	10–1F	S2-0 to S2-2	↕
20	20–2F	S2-0 to S2-3	↕
30	30–3F	S2-0 to S2-4	↕
40	40–4F	S2-0 to S2-5	↕
50	50–5F	S2-0 to S2-6	↕
60	60–6F	S2-0 to S2-7	↕
70	70–7F	S2-0 to S2-8	29–30
80	80–8F	S2-0 to S2-1	28–29
90	90–9F	S2-0 to S2-2	↕
A0	A0–AF	S2-0 to S2-3	↕
B0	B0–BF	S2-0 to S2-4	↕
C0	C0–CF	S2-0 to S2-5	↕
D0	D0–DF	S2-0 to S2-6	↕
E0	E0–EF	S2-0 to S2-7	↕
F0	F0–FF	S2-0 to S2-8	28–29

S2-0 to S2-9 is OFF.

2.2 XACK/ GENERATION LOGIC

The XACK/ generation logic consists of two 74LS193 counters, a five-pair time base jumper pad, a 74S74 flip-flop and various gating circuits, as shown on sheet 1 of the board schematic, Figure 9.

The XACK/ generation logic is responsible for acknowledging, at the proper time, all I/O read (IORC/) and I/O write (IOWC/) commands directed to the DMA Board. The 9.8304 MHz common clock pulse (CCLK/) can be applied to the down-count input of the first of two 74LS193 counters or can feed the up-count input of the second counter directly (jumper pair 25-26 must be connected). If jumper pair 26-27 is not connected, the first counter divides the CCLK/ pulse by two (200 ns), four (400 ns), eight (800 ns) and 16 (1600 ns), at its QA, QB, QC and QD outputs, respectively. Each counter output is tied to one-half of a jumper pair. The particular jumper pair which is connected determines the frequency of the up-count input to the second counter. This counter is pre-loaded to 12_{10} (1100_2). The occurrence of IORC/ or IOWC/ causes the LD input to the counter to go false, allowing the count sequence to begin. Two counts later, the QB output from the counter goes high.

If IORC is true (i.e., if it is an I/O read operation), the D-input to a 74S74 latch goes high when QB goes high. On the next count pulse, the latch is clocked set and XACK/ goes true. The occurrence of XACK/ disables all further counting. XACK/ is available at pin P1-23 and remains true until IORC/ goes false.

If an I/O write operation is in progress, instead of a read, the D-input to the XACK/ latch does not go true until the QD counter output goes true; that is, eight count pulses after IOWC/ appeared. As before, the XACK/ is clocked set on the leading edge of the next count pulse. XACK/ goes true and

Table 2

I/O PORT ADDRESS DECODING

PORT ADDRESS (HEX)	I/O COMMAND	DECODER LOCATION	SIGNAL (PIN)	DESCRIPTION
BASE+0	IOWC/ (output)	A61	OUTPORT 0/ (J2-56)	Output strobes to external devices
BASE+1			OUTPORT 1/ (J2-54)	
BASE+2			OUTPORT 2/ (J2-52)	
BASE+3			OUTPORT 3/ (J2-50)	
BASE+4			OUTPORT TAG/ (J2-10)	Not Used
BASE+5				
BASE+6				
BASE+7		A61		
BASE+8		A48	SET INT/	Set the "SET INT" latch (also sets the INT latch if DMA not busy
BASE+9			RESET INTERRUPT/	Clears interrupt, DMA busy and SET INT latches
BASE+A			LOAD CNTRL REGISTER/	Load control register
BASE+B			LOAD TAG REGISTER/	Load tag register
BASE+C			OUTP LEN LSBY/	Load least significant byte of length register
BASE+D			OUTP LEN MSBY/	Load most significant byte of length register
BASE+E			OUTP MA LSBY/	Load LSBY of memory address register
BASE+F	IOWC/ (Output)	A48	OUTP MA MSBY/	Load MSBY of memory address register
BASE+0	IORC/ (input)	A49	INPORT O/ (J2-16)	Input strobes to external devices
BASE+1			INPORT 1/ (J2-18)	
BASE+2			INPORT 2/ (J2-20)	
BASE+3			INPORT 3/ (J2-22)	
BASE+4			RD LEN LSBY/	Read LSBY of length register
BASE+5			RD LEN MSBY/	Read MSBY of length register
BASE+6			RD DMAC STAT/	Read DMA status word
BASE+7		A49	INPORT RAG/ (J2-12)	Input strobe to external devices
BASE+8				Not Used
BASE+9				
BASE+A				
BASE+B				
BASE+C				
BASE+D				
BASE+E				
BASE+F	IORC/ (input)			

disables any further counting. XACK/ is available at pin P1-23 and remains true until IOWC/ goes false.

Figure 2 illustrates XACK/ timing for each of the five time base jumper pairs, during both I/O read and write cycles.

When the DMA Board, as bus master, is transferring a data word to or from memory, the memory board returns an XACK/ signal in response to the memory read (MRDC/) or write (MWTC/) command generated by the DMA Board. This XACK/ signal is received at pin P1-23 and made available to the DMA transfer control logic (see Section 2.10).

2.3 CONTROL REGISTER

The control register is a 3404 six-bit, high-speed latch, shown on sheet 2 of the board schematic, Figure 9.

The control register is normally loaded prior to the beginning of a DMA operation. When the CPU executes an I/O write instruction to port "BASE+ A_{16}", the DMA Board's address decoding logic generates the LOAD CNTRL REGISTER/ signal which is gated through to the write enable inputs on the 3404 six-bit latch. Data bits 0–3 from the CPU are buffered and inverted in the bus in/out logic and applied to four of the control register inverting data inputs. Data bits 4 and 5 are applied directly to the two remaining 3404 inverting data inputs. Bit definitions for the control register are as follows:

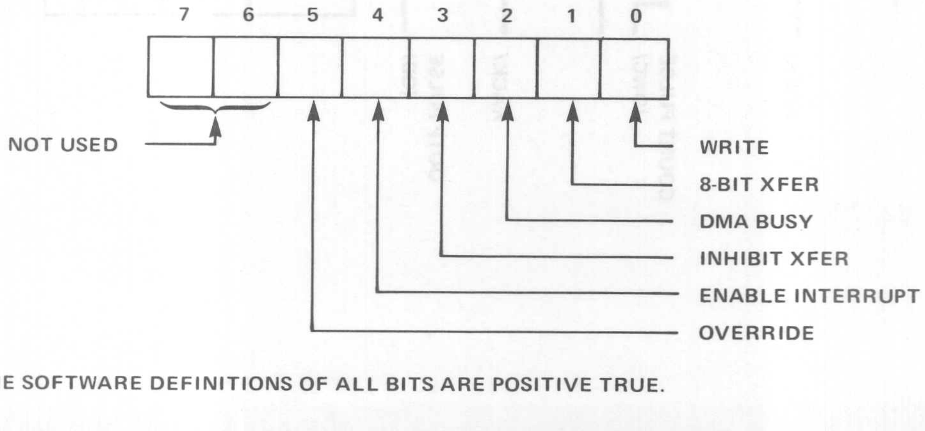
When the WRITE/ signal is true (low), it indicates that data is to be transferred from an external device to memory (WRITE MODE). When WRITE/ is high, it indicates that data is to be transferred from memory (READ MODE) to an external device.

When the 8 BIT XFER/ signal is true (low), it indicates that 8-bit data words are to be transferred. When 8 BIT XFER/ is high, it indicates that 16-bit data words are to be transferred.

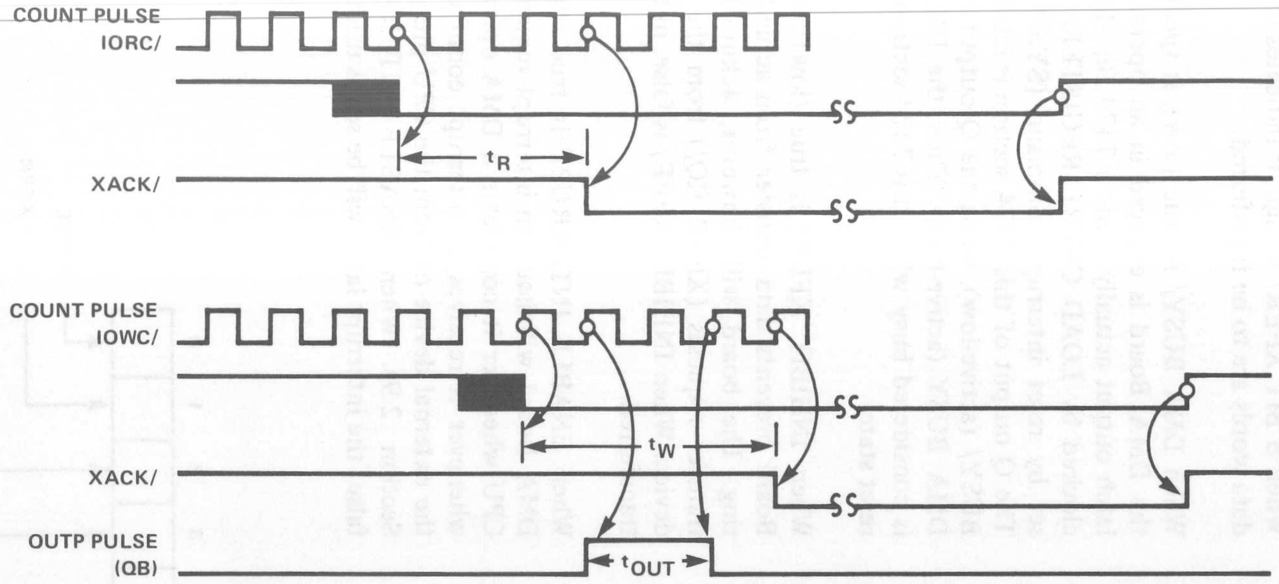
When DMA BUSY/ is true (low), it specifies that the DMA Board is engaged in an operation. The latch output actually feeds a 7474 flip-flop that is clocked by LOAD CNTRL REGISTER/ and preset by reset interrupt command (SYSINTRST/). The Q output of this 7474 section is labeled DMA BUSY/ (active-low), while the Q output is labeled DMA BUSY (active-high). Thus, the DMA Board is considered busy when this 7474 section is in the reset state.

When INHIBIT XFER/ is true (low), the DMA Board prevents data transfer from actually occurring. The board will, however, acknowledge all transfer requests (XFER RQ/) from the external device. When INHIBIT XFER/ is false, data will be transferred.

When ENABLE INTERRUPT is true (high), the DMA Board will issue an interrupt request to the CPU whenever it completes a DMA operation, or whenever it receives an interrupt command from the external device or from the CPU program (see Section 2.9). When ENABLE INTERRUPT is false, the interrupt latch will be set as usual, but no



NOTE: THE SOFTWARE DEFINITIONS OF ALL BITS ARE POSITIVE TRUE.



TIME BASE JUMPER PAIR CONNECTED	COUNT PULSE PERIOD (ns)	t_R (I/O READ)	t_W (I/O WRITE)	t_{OUT} (I/O WRITE)
25–26	100	200 ns	800 ns	200 ns
3–4	200	400 ns	1.6 μs	400 ns
1–2	400	800 ns	3.2 μs	800 ns
5–6	800	1.6 μs	6.4 μs	1.6 μs
7–8	1600	3.2 μs	12.8 μs	3.2 μs

Figure 2. XACK/ Generation Timing (Nominal)

interrupt request will be issued to the CPU. The status of the interrupt latch can be read by the CPU program, however.

Normally, the DMA Board requests use of the system bus for each word transferred. If, however, uninterrupted use of the bus is required, the CPU can set the override bit in the control register. If the override bit is set and the interrupt latch is not set, the OVERIDE/ signal will be true (low). When OVERIDE/ is true, the bus interface logic is prevented from relinquishing control of the system bus (see Section 2.11).

2.4 LENGTH REGISTER

The length register consists of four 74LS193 4-bit counters, as shown on sheet 3 of the board schematic, Figure 9. Prior to the beginning of a DMA transfer operation, the length register is loaded with the initial 16-bit word count (i.e., the total number of data words to be transferred).

The least significant byte of this initial word count value is output onto the system data bus when the CPU executes an I/O write instruction directed to port "BASE+C". The address decoding logic generates the OUTP LEN LSBY/ signal which is applied to the lead inputs on the first two 74LS193 counters. The most significant byte of the initial word count value is output when the CPU executes an I/O write instruction to port "BASE+D". The OUTP LEN MSBY/ signal is applied to the load inputs on the other two 74LS193 counters.

If the length register contents do not equal zero, the contents are decremented before each data word is transferred. The LENGTH CNT signal, generated by the DMA transfer control logic, is applied to the down-count input on the first counter. The borrow output from each counter is, in turn, applied to the down-count input of the next counter; essentially creating a 16-bit binary counter out of the four 4-bit counters. The borrow output from the fourth counter constitutes the (LEN REG=Ø)/ signal, which when low (active) indicates that all of the specified data words have been transferred.

The outputs of the four counters are applied to two 74S258 eight-to-four multiplexers. When the CPU executes an I/O read instruction to port "BASE+4", a low level on the RD LEN LSBY/ signal (active) from the address decoding logic enables the least significant byte of the 16-bit word count value (i.e., the contents of the first two counters) through the two multiplexers and onto the system data bus (DATØ/–DAT7/). Similarly, when the CPU executes an I/O read instruction to port "BASE+5", the low (active) level on the RD LEN MSBY/ signal and the high (inactive) level on the RD LEN LSBY/ signal, enable the contents of the other two counters (the most significant byte) through the two multiplexers and onto the system data bus (DATØ/–DAT7/).

2.5 MEMORY ADDRESS REGISTER

The memory address register, like the length register, consists of four 74LS193 4-bit counters, as shown on sheet 3 of the board schematic, Figure 9. Prior to the beginning of a DMA transfer operation, the memory address register is loaded with the 16-bit address of the first location to be accessed.

The least significant byte of the address is output onto the system data bus (not the system address bus) when the CPU executes an I/O write instruction to port "BASE+E_{16}". The address decoding logic generates the OUTP MA LSBY/ signal which is applied to the load inputs on the first two 74LS193 counters. The most significant byte of the memory address is output when the CPU executes an I/O write instruction to port "BASE+F_{16}". The OUTP MA MSBY/ signal is applied to the load inputs on the other two 74LS193 counters.

If the contents of the length register do not equal zero, the contents of the memory address register are incremented before each data word is transferred. The MEMORY CNT signal, generated by the DMA transfer control logic, is applied to the up-count input on the bottom counter. The carry output from each counter is, in turn, applied to the up-count input of the counter above it. The carry output from the top counter is not used.

The outputs of the four counters feed sixteen 8098 inverters which, when enabled by the SELECTED/

signal from the bus interface control logic (indicating that the DMA Board has control of the system bus), drive the 16 address bits to memory on the system address bus. ADRØ/–ADRF/ (pins P1-43 to P1-58).

These address bits are accompanied by a memory read (MRDC/) or write (MWTC/) command, generated in the DMA transfer control logic. MRDC/ or MWTC/ inform the memory when the address on the bus is valid, as well as specifying the transfer direction. Timing for MRDC/ and MWTC/ is provided in Section 2.10.

2.6 TAG REGISTER

The tag register consists of four 3404 high-speed latches, as shown on sheet 2 of the board schematic, Figure 9.

The tag register is loaded from bits 0–3 of the system data bus (DATØ/–DAT3/) when the CPU executes an I/O write instruction directed to port "BASE+B_{16}". The LOAD TAG REGISTER/ signal from the address decoding logic enables the write inputs on the 3404 latches. The four outputs from the tag register, TAGØ/ (pin J2-40), TAG1/ (pin J2-38), TAG2/ (pin J2-36) and TAG3/ (pin J2-24), are driven by four 7437 NAND gates to the external devices. The tag register is cleared by the system reset signal (SYS RST/), provided that system data bus lines DATØ/–DAT3/ are all false (high).

The tag register can be used at the designer's discretion. For example, each bit in the register can be used as a command line to the external devices. Or, the tag register can be used to expand the maximum number of ports supported by the DMA Board. In this case, the four tag lines could "steer" the data, output with any of the OUTPORT X/ strobes (see Section 8.2.1), to one of 16 devices.

2.7 STATUS LOGIC

The status logic consists of two 74S257 multiplexers that gate four internal and four external status lines onto the system data bus during read status operations, as shown on sheet 3 of the board schematic, Figure 9.

When the CPU executes an I/O read instruction directed to port "BASE+6", the RD DMAC STAT/ and INPUT EN2/ signals (generated in the address decoding logic), allow the status lines through the two multiplexers and onto the appropriate system data bus lines, as listed in Table 3.

The status informatin on the data bus remains stable until the I/O read command, IORC/, goes false. Figure 3 illustrates read status timing.

Table 3

STATUS LINES

	STATUS LINE	(PIN)	SOURCE	DATA BUS LINE	(PIN)
External	STATUS0/	(J1-48)	External Device	DAT0/	(P1-73)
	STATUS1/	(J1-46)		DAT1/	(P1-74)
	STATUS2/	(J1-44)		DAT2/	(P1-71)
	STATUS3/	(J1-42)		DAT3/	(P1-72)
Internal	SET INT/		DMA interrupt logic	DAT4/	(P1-69)
	MEM WRT/		DMA transfer ctl logic	DAT5/	(P1-70)
	INTR STAT/		DMA interrupt logica	DAT6/	(P1-67)
	DMA BUSY/		Control register	DAT7/	(P1-68)

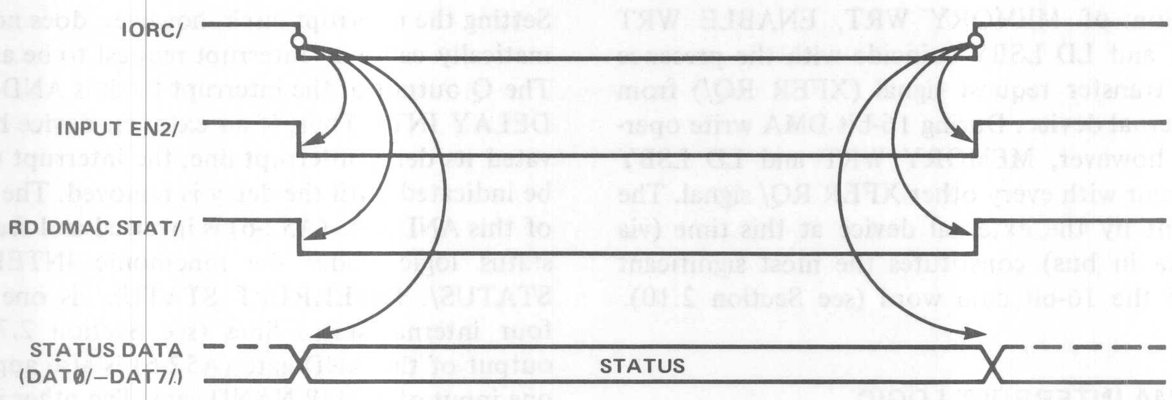


Figure 3. Read Status Timing

2.8 BUS IN/OUT LOGIC

The bus in/out logic routes data between the data in bus (from external devices), the data out bus (to external devices), the bidirectional system data bus and various registers within the DMA Board. This block consists of an 8212 8-bit I/O port device, sixteen 3404 latches, two 74157 multiplexers and various gating circuits as shown on sheet 3 of the board schematic, Figure 9.

Control or address information that is output by the CPU on the system data bus and intended for internal DMA Board registers is inverted and latched into 3404 circuits (by the IOWC/ signal), and applied to two 74157 multiplexers. These multiplexers allow the DMA Board to accept either 8-bit or 16-bit words. When the RD MSBY signal from the DMA transfer control logic is low (false), the low-order data lines (DATØ–DAT7) are multiplexed through to the internal registers. When RD MSBY is high (true), the high-order data lines (DAT8–DATF) are multiplexed through. The DMA transfer control logic (see Section 2.10) alternates the level on RD MSBY when a 16-bit operation is indicated by the contents of the control register. You will notice on sheet 3 of the board schematic that data lines DATØ/–DAT5/ are tapped off the sytem bus prior to the 3404 circuits and applied directly to the control and tag registers.

Control or address information that is output by the CPU and intended for control or address registers in an external device are inverted, buffered and multiplexed through the bus in/out logic as described above. The I/O write command (IOWC/) enables eight 7437 NAND gates that drive the byte to the external devices on the data out bus (DATA OUTØ/–DATA OUT7/). One of the OUTPORT strobes, generated in the address decoding logic, informs the appropriate external device that it is to accept the byte in the data out bus.

DMA data that is being transferred from memory to an external device also follows the same path described in the preceding paragraphs. In this case, however, it is the GATE READ DATA/ signal from the DMA transfer control logic which enables the eight 7437 NAND gates that drive the data out bus, and it is the XFER ACKNOWLEDGE/ signal (pin J1-4), also from the DMA transfer control logic (see Section 2.10), that strobes the data byte into the external device which issued a transfer request signal (XFER REQ/; pin J1-1). XFER REQ/ is issued for each byte to be transferred to/from an external device. XFER REQ/ also enables the GATE READ DATA/ signal mentioned above.

DMA data that is being transferred from an external device to memory is passed to the DMA Board on the data in bus (DATA INØ/–DATA IN7/). If the board is operating in the 8-bit mode (as defined by the contents of the control register) or if the data byte is the least significant byte of a 16-bit transfer, the data from the data in bus is applied to the eight inputs of an 8212 I/O port device. The MEMORY WRT, ENABLE WRT DATA/ and LD LSBY signals from the DMA transfer control logic (see Section 2.10) enable the 8212 section and are driven over the low-order lines of the system data bus (DATØ/–DAT7/). The generation of MEMORY WRT, ENABLE WRT DATA/ and LD LSBY coincide with the presence of the transfer request signal (XFER RQ/) from the external device. During 16-bit DMA write operations, however, MEMORY WRT and LD LSBY only occur with every other XFER RQ/ signal. The data sent by the external device at this time (via the data in bus) constitutes the most significant byte of the 16-bit data word (see Section 2.10).

2.9 DMA INTERRUPT LOGIC

The DMA interrupt logic consists of a 7474 D-type flip-flop (interrupt latch), a user installed jumper (interrupt level select) and assorted gating circuits as shown on sheet 1 of the board schematic, Figure 9.

The interrupt latch can be set in three different ways:

- The CPU issues a set interrupt command by executing an I/O write instruction to port "BASE+8". (This also sets the SET INT latch.)
- A peripheral device activates its external interrupt line, EX INTERRUPT/ (pin J1-3).
- The DMA Board completes a transfer operation (i.e., when the length register is decremented to zero).

When the CPU executes an I/O write instruction to port "BASE+8", the address decoding logic generates the SET INT/ signal. SET INT/ is saved in the S-R latch, composed of two 7400 OR gates. The output of the S-R latch, SET INT STATUS/ constitutes one of the four internal status lines that can be read by the CPU (see Section 2.7). SET INT/ is also inverted and NANDed with DMA BUSY/. If DMA BUSY/ is high (i.e., the DMA Board is not busy), the NAND gate output pre-sets the interrupt latch (A65-10).

If an external device activates its EX INTERRUPT/ line, the interrupt latch is pre-set.

If the DMA Board is currently involved in an operation (i.e., DMA BUSY is true) when the (LEN REG=∅)/ signal goes true, the interrupt is clocked to the set state.

Setting the interrupt latch, however, does not automatically cause an interrupt request to be asserted. The Q output of the interrupt latch is ANDed with DELAY INT/. Thus, if an external device has activated its delay interrupt line, the interrupt will not be indicated until the delay is removed. The output of this AND gate (A52-6) is inverted and fed to the status logic under the mnemonic INTERRUPT STATUS/. INTERRUPT STATUS/ is one of the four internal status lines (see Section 2.7). The output of the AND gate (A52-6) is also applied to one input of a 7438 NAND gate. The other input is the ENABLE INTERRUPT line from the DMA Board's control register (see Section 2.3). If interrupts are enabled, the open collector output of A37-3 is connected to wire wrap post S1-0 and S1-9. The first eight wire wrap posts (S1-1 through S1-8) are tied to the eight interrupt level request lines (INT∅/-INT7/); position 9 is off. The interrupt request to the CPU will be asserted on the level selected by the position of the jumper that is installed.

After the CPU services the interrupt request, it will issue a reset interrupt request command (RESET INTERRUPT/) by executing an I/O write instruction to port "BASE+9". RESET INTERRUPT/ resets the interrupt latch, the SET INT latch and the BYTE CNTR latch in the DMA transfer control logic (see Section 2.10). RESET INTERRUPT/ also clears the DMA BUSY/ signal, shown on sheet 2 of the board schematic. In addition, the RESET INTERRUPT/ signal is made available to the external devices (via pin J1-14).

2.10 DMA TRANSFER CONTROL LOGIC

The DMA transfer control logic is responsible for the "handshaking" exchanges with the peripheral device and memory during all DMA operations. The DMA transfer control logic consists of two 7474 D-type flip-flops, two 74S74 D-type flip-flops, one 74109 J-$\overline{K}$ flip-flop and assorted gating circuits, as shown on sheet 1 of the board schematic, Figure 9.

Once a DMA transfer operation has been initiated, the external device will activate its transfer request line (XFER RQ/) whenever it is ready to send a data byte to memory or to receive a byte from memory. The direction of data flow is indicated by the level on the XFER DIR IN line (pin J1-6) from the external device. A high level on XFER DIR IN specifies that the device expects to receive a data byte from memory; a low level specifies that the device has data to be written to memory. If the XFER DIR IN is left open (high), bit 0 (WRITE) in the control register specifies the direction of data flow.

XFER RQ/ is received at pin J1-1, inverted, buffered and applied to the clock input of the BUS XFER RQ/ latch. This 7474 latch will be clocked to the set state *unless* one of the following is true:

(1) the contents of the control register specify that the current DMA operation is not to involve a data transfer (i.e., if INHIBIT XFER/ from the control register is true);

(2) the contents of the length register are equal to zero (i.e., if (LEN REG=Ø)/ is true);

(3) the first (least significant) byte of a 16-bit word is being input from an external device; or

(4) the second (most significant) byte of a 16-bit word is being output to an external device.

The $\overline{Q}$ output of the 7474 section, BUS XFER RQ/, is fed to the bus interface logic (see Section 2.11).

When one of the first two above-mentioned conditions is true (INHIBIT XFER/ or (LEN REG=Ø)/), BUS XFER RQ/ is not generated because no transfer is to occur. However, if any of the four conditions are true when XFER RQ/ occurs, the XFER ACKNOWLEDGE (another 7474 section) latch will be clocked to the set state. When this latch is set, XFER ACKNOWLEDGE/ is asserted at pin J1-4. During all other DMA transfer cycles, the XFER ACKNOWLEDGE/ latch is clocked by RQ ACK/. RQ ACK/ is generated when the DMA board receives the transfer acknowledge signal (XACK/), returned by memory in response to a memory read (MRDC/) or write (MWTC/) command. Thus, the external device's transfer request is always acknowledged, whether a transfer is actually performed or not. The XFER ACKNOWLEDGE/ latch is pre-reset when XFER RQ/ goes false.

During 16-bit DMA transfer to memory, BUS XFER RQ/ is generated when the external device activates XFER RQ/ in order to send the *second* (most significant) byte to the DMA Board. The entire 16-bit word (two bytes) is then transferred in parallel to memory. The first (least significant) byte was latched into an 8-bit 8212 device (A17) on the previous XFER RQ/.

During 16-bit DMA transfers from memory, BUS XFER RQ/ is generated when the external device activates XFER RQ/ in order to receive the *first* (least significant) byte of data. Memory, in this case, will actually send both data bytes at once. However, the two data bytes will be transferred to the external device separately.

This byte selection process is controlled by the BYTE CNTR latch (a 72S74 latch shown on sheet 1 of the board schematic). During 8-bit transfer operations, the 8-BIT XFER/ signal from the control register keeps the BYTE CNTR latch pre-set (A9-4). During 16-bit transfers, however, the 8-BIT XFER/ signal is false and the latch toggles on the trailing (positive-going) edge of each XFER RQ/ pulse. When the BYTE CNTR latch is clocked to the set state it indicates that the most significant (second) data byte is to be transferred to/from the external device. RD MSBY is generated when the BYTE CNTR latch is set and XFER REQ/ goes true. When the BYTE CNTR latch is clocked to the reset state or when 8-BIT XFER/ is true, it indicates that the first, least significant (or only) data byte is to be transferred to/from the external device. LD LSBY is generated when XFER RQ/ goes true, unless the length register equals zero. RD MSBY and LD LSBY are used to route the data through the bus in/out logic (see Section 2.8). The BYTE CNTR latch is pre-reset by a RESET INTERRUPT command (see Section 2.9). This ensures that the BYTE CNTR latch is in the proper state for a 16-bit transfer which may follow.

The DMA transfer control logic also generates the read (MRDC/) and write (MWTC/) commands for memory, as well as various read and write signals that are used internally. The Q output of the BUS XFER RQ/ latch (A66-9) feeds two NAND gates. The other input to the first 7400 gate is the result of ANDing XFER DIR IN and WRITE/. This resultant ANDed output is referred to as MEM WRT/. Both XFER DIR IN and WRITE/ are high during transfers from memory (memory read); thus, MEM WRT/ is false (high) and enables the first 7400 gate (A8-4). MRD RQ/ is the output from this gate (it is true during memory read operations). The AND of XFER DIR IN and WRITE/ is also inverted (referred to as MEMORY WRT; active-high) and applied to the other input of the second 7400 NAND gate. MWT RQ/ is the output from this gate (it is true during memory write operations). If the DMA Board's bus interface logic has gained control of the system bus (see Section 2.11), MRD RQ/ and MWT RQ/ are gated through to pins P1-19 and P1-20 and driven to memory as MRDC/ and MWTC/, respectively. MRD RQ/ is also available to the bus in/out logic, where it causes the data being read from memory to be latched into 3404 circuits (see Section 2.8).

When MEMORY WRT is true (high) it is also NANDed with the DMA SELECTED signal from the bus interface logic to produce ENABLE WRT DATA/. When MEM WRT/ is false (high), it is NANDed with XFER RQ to produce GATE READ DATA/. MEM WRT/ is also inverted and made available to the external device as XFER DIR OUT (pin J1-2). XFER DIR OUT is high during memory read operations and low during memory write operations. MEMORY WRT, MEM WRT/, ENABLE WRT DATA/ and GATE READ DATA/ are all made available to the bus in/out logic, where they are used to gate data on or off the system data bus (see Section 2.8).

Figures 4 and 5 illustrate the relative timing between DMA transfer control logic signals during memory read cycles (memory-to-external device) for 8 and 16-bit transfers, respectively. Figures 6 and 7 illustrate timing during memory write cycles (external device-to-memory) for 8 and 16-bit transfers, respectively.

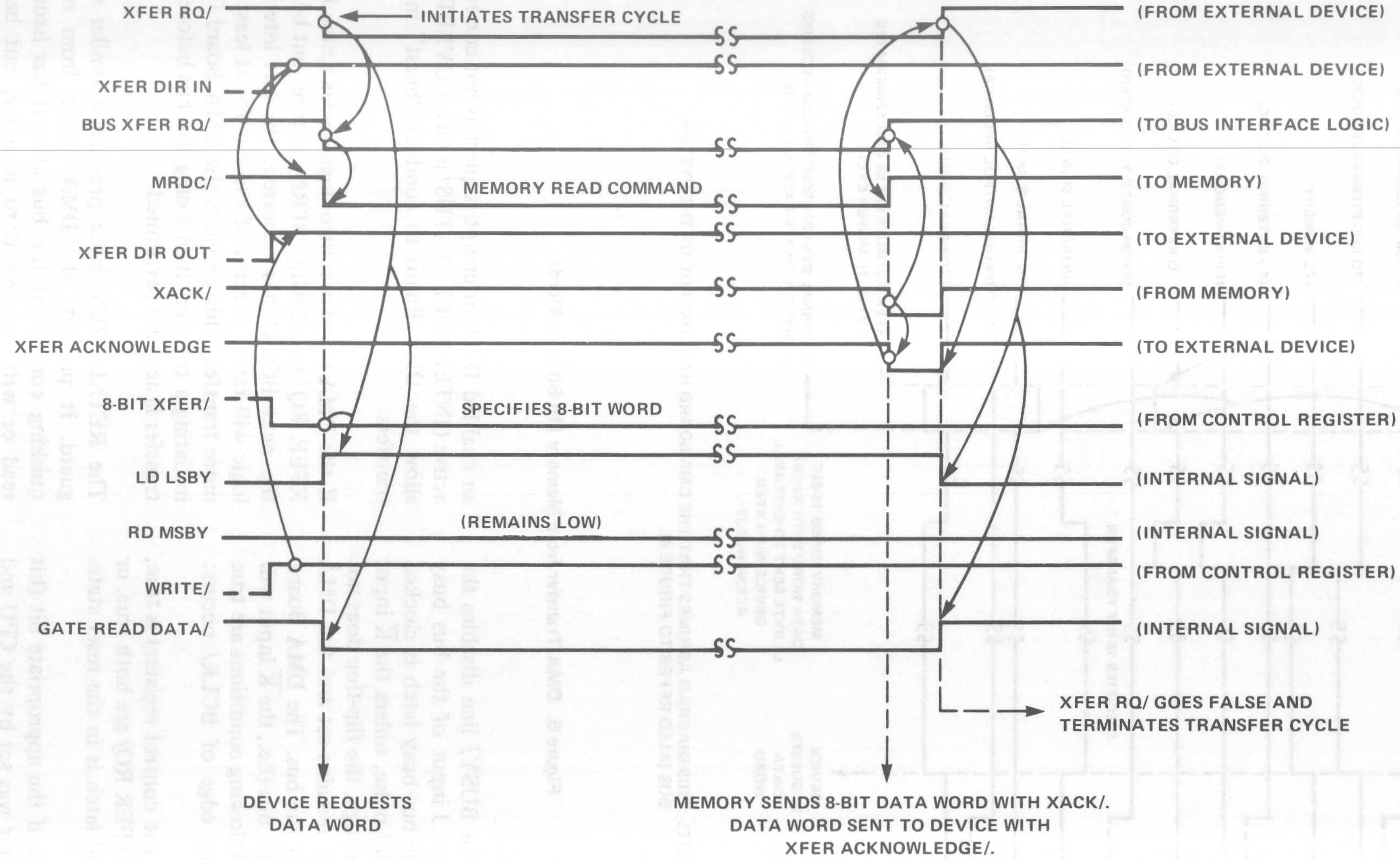
2.11 BUS INTERFACE LOGIC

The bus interface logic is responsible for requesting and maintaining control of the system bus when the DMA Board is conducting a data transfer. The bus interface logic consists of two 74S74 D-type flip-flops, two 74109 J-$\overline{K}$ flip-flops and assorted gating circuit, as shown on sheet 1 of the board schematic, Figure 9.

When the DMA transfer control logic determines that a data transfer is to be performed by the DMA Board, it issues the BUS XFER RQ/ signal to the bus interface logic. BUS XFER RQ/ is inverted and applied to the D input of the bus request latch (74S74 flip-flop). The bus request latch is subsequently clocked to the set state, on the positive-going edge of the next bus clock pulse (BCLK/). The $\overline{Q}$ output (BREQ/) is asserted at pin P1-18. BREQ/ specifies that the DMA Board requires use of the system bus. Logic on some other module will resolve all bus requests according to priority. In the INTELLEC® Microcomputer Development System, the Front Panel Control Board provides an eight-level, parallel bus priority scheme.

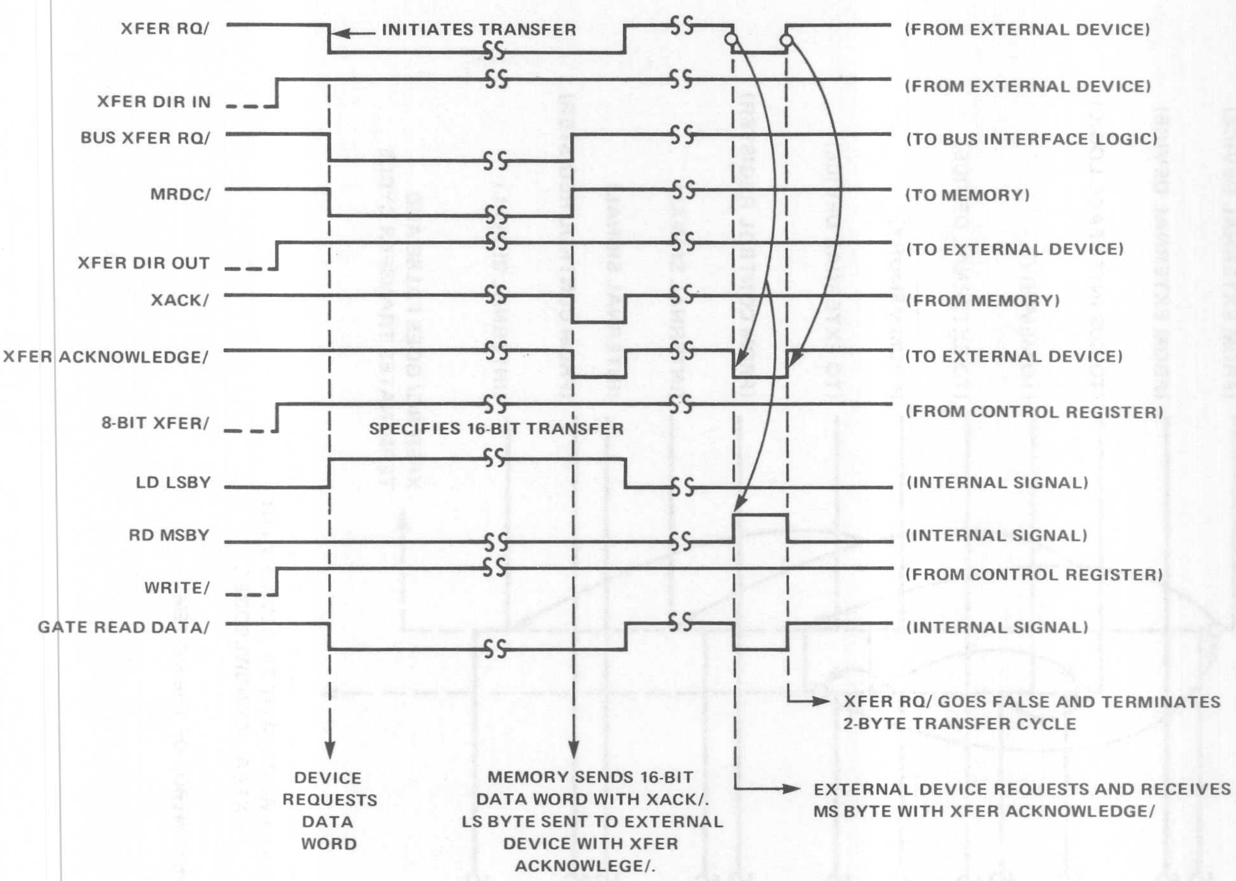
When the bus is available for use by the DMA Board, the BPRN/ signal (pin P1-15) will be true. BPRN/ is inverted and applied to one of three inputs on a 74H11 AND gate (A25-2). The other two inputs are supplied by the Q output from the bus request latch and the bus busy signal (BUSY/). Thus, the 74H11 gate will be activated if PBRN is true, the DMA Board is currently requesting use of the bus (BREQ is true) and the bus is not already in use (BUSY/ is false). The output of this 74H11 section (A25-12) feeds the J input of the bus busy latch (a 74109 section). The bus busy latch is clocked set by the positive-going edge of the next bus clock pulse (BCLK/). The Q output (DMA SELECTED) informs the DMA transfer control logic that the board now has control of the system bus. The $\overline{Q}$ output (EN MEM ADR/) enables the 74125 driver which asserts the bus busy signal (BUSY/) at pin P1-17. EN MEM ADR/ also enables the sixteen 8098 inverters which drive the contents of the memory address register on the system address bus (ADRØ/–ADRF/). This address is available to memory. DMA SELECTED is also fed to the J and $\overline{K}$ inputs on another 74109 flip-flop. The next BCLK/ pulse clocks this section to the set state. The $\overline{Q}$ output, in turn, enables the two 74125 drivers which enable the memory read (MRDC/) or write (MWTC/) command onto the system bus (pins P1-19 and P1-20, respectively). Thus, the appropriate memory command (see Section 2.10) is gated onto the system bus one bus clock period after the memory address is enabled.

Normally, the DMA Board will only retain control of the bus for the transfer of one data word. An



NOTE: THIS DIAGRAM ASSUMES THAT THE DMA BOARD HAS CONTROL OF THE SYSTEM BUS (ALSO REFER TO FIGURE 8).

Figure 4. DMA Transfer from Memory (8-Bit Data Word)



NOTE: THIS DIAGRAM ASSUMES THAT THE DMA BOARD HAS CONTROL OF THE SYSTEM BUS (ALSO REFER TO FIGURE 8).

Figure 5. DMA Transfer from Memory (16-Bit Data Word)

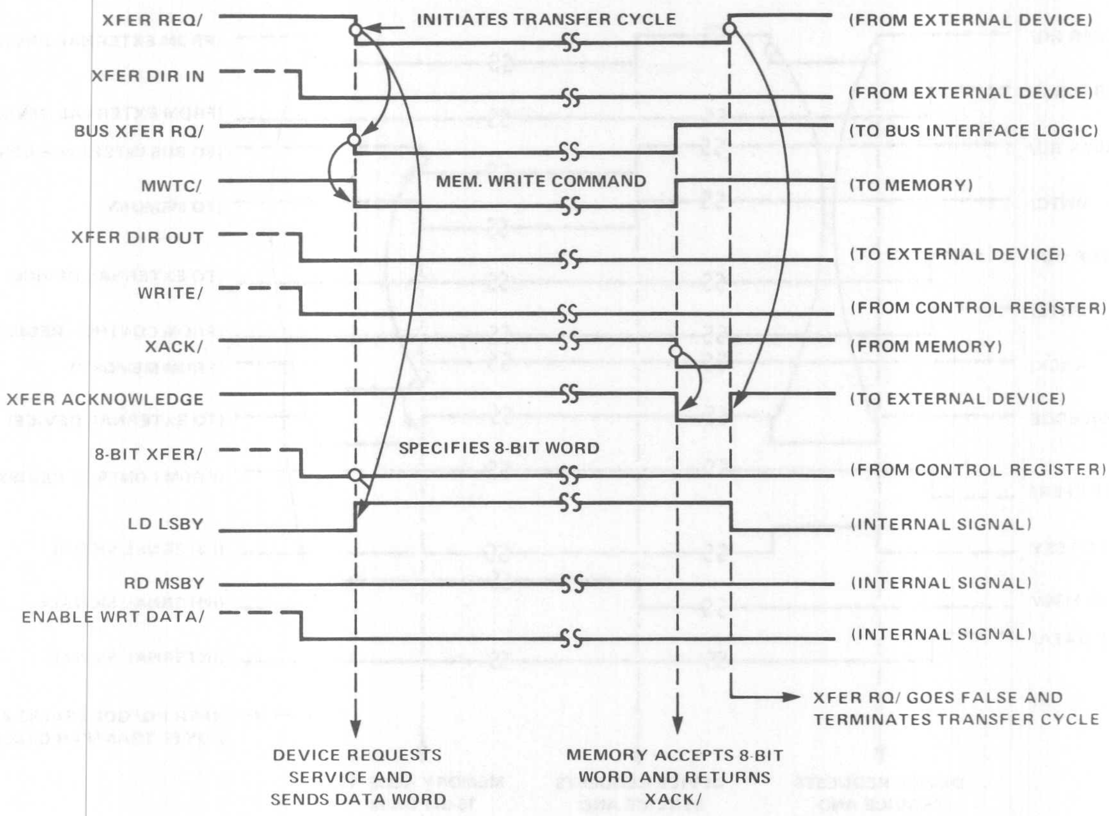
active (low) level on the BUSY/ line disables the 74H11 gate, feeding the J input of the bus busy latch; consequently, the bus busy latch is clocked reset by the next BCLK/ pulse, unless the $\overline{K}$ input is high. If the $\overline{K}$ input is high, the flip-flop does not change states; that is, it remains set and the DMA Board retains control of the bus. The DMA Board will retain control of the bus (i.e., the $\overline{K}$ input will be high) if any of the following conditions are true when the positive-going edge of BCLK/ occurs:

- OVERIDE (from the control register) is true,
- BPRN/ and BUS XFER RQ/ are both true, or
- the RETAIN BUS/ latch is in the reset state.

OVERIDE/ will be true if the appropriate bit (bit 5) in the control register was set by the CPU and an enabled DMA interrupt request is not currently active (INTERRUPT STATUS/ is false). OVERIDE/ allow the DMA Board to conduct "burst" mode transfers.

If the DMA transfer control logic issues a new BUS XFER RQ/ signal while BPRN/ is true, but before the next BCLK/ has occurred, the bus interface logic will retain control of the bus for at least one more transfer. This prevents the DMA Board from initiating a transfer, then losing the bus before the transfer is actually performed.

The RETAIN BUS/ latch provides a similar safeguard. It prevents the DMA Board from relinquishing control of the bus after it has issued a read or write command to memory, but before

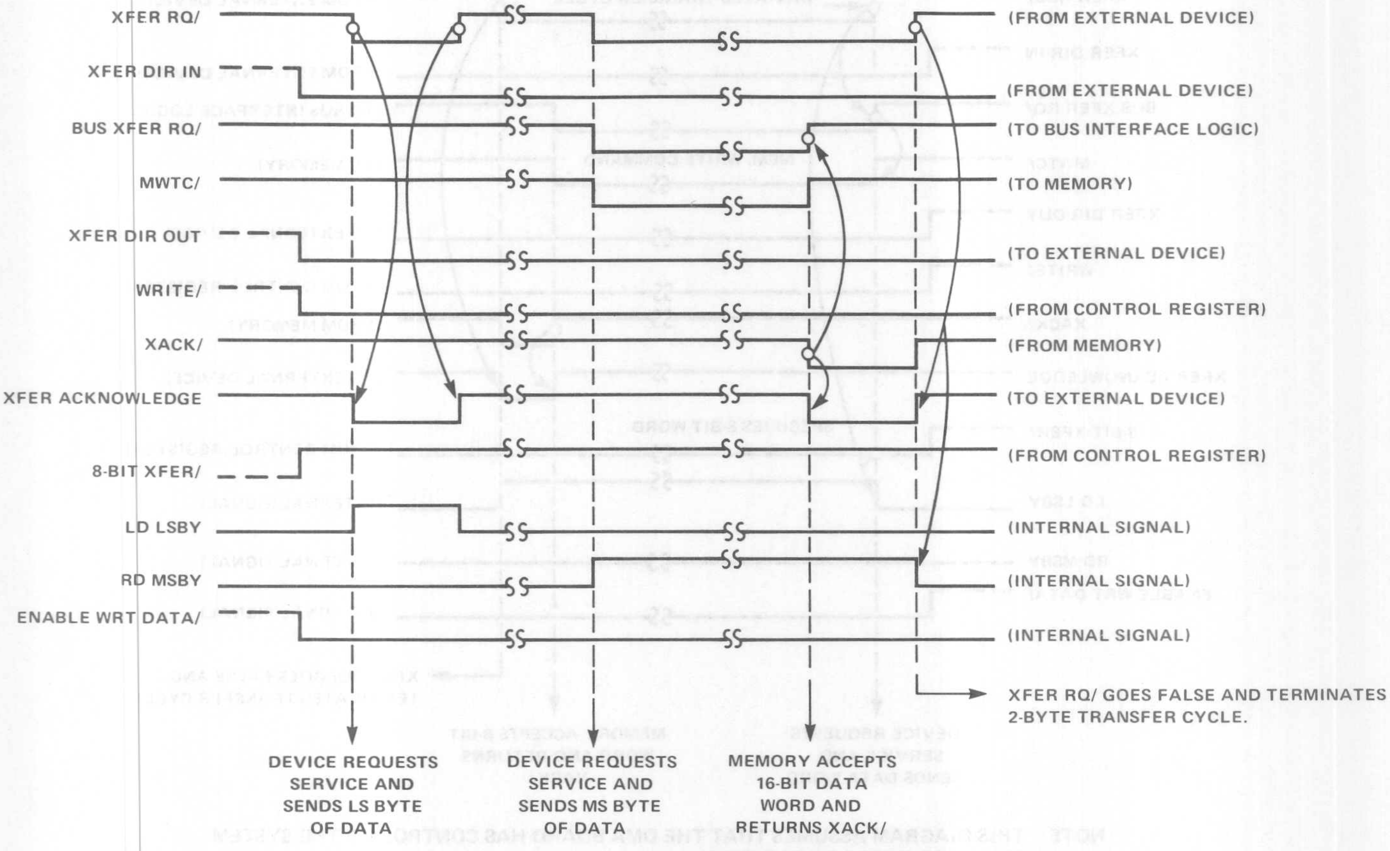


NOTE: THIS DIAGRAM ASSUMES THAT THE DMA BOARD HAS CONTROL OF THE SYSTEM BUS (ALSO REFER TO FIGURE 8).

Figure 6. DMA Transfer to Memory (8-Bit Data Word)

memory has actually completed the transfer. If the DMA Board has control of the bus (SELECTED/ is true) the RETAIN BUS/ latch is clocked when the memory read (MRD RQ/) or memory write (MWT RQ/) request signal (from the DMA transfer control logic) or the bus request signal (BREQ/) is generated. The latch will be clocked to the reset state if OVERIDE/ or BPRN/ is true. Otherwise, the latch will remain reset. The Q output (low when the latch is reset) enables the MRD RQ/ or MWT RQ/ signal through to the MRDC/ or MWTC/ driving circuit. The Q output also asserts a high level on the K input of the bus busy latch (as mentioned above), forcing the bus interface logic to retain control of the bus. When MRD RQ/, MWT RQ/ and BREQ/ are all false (high), the RETAIN BUS/ latch is pre-set.

The DMA Board's bus interface logic is primarily controlled by the state of the bus priority in (BPRN/) signal. When BPRN/ is true, the board can gain or retain control of the bus, and when BPRN/ is false, the board will relinquish control of the bus, unless override has been invoked. BPRN/ may be generated by a central *parallel* priority network; in the INTELLEC® system such a network is included on the Front Panel Control Board. BPRN/ may also be generated and transmitted in *serial*. In such a case, BPRN/ is captured by the highest priority board requiring control of the bus. Those boards that do not require the bus accept BPRN/ and pass BPRO/ (bus priority out) on to the next board on the bus (via pin P1-16). Thus, a board's priority in a serial network is dependent on its relative position on the bus.



NOTE: THIS DIAGRAM ASSUMES THAT THE DMA BOARD HAS CONTROL OF THE SYSTEM BUS (ALSO REFER TO FIGURE 8).

Figure 7. DMA Transfer to Memory (16-Bit Data Word)

BPRO/ is generated if BPRN/ is true and BREQ is false (i.e., the DMA Board is not requesting use of the system bus).

Figure 8 illustrates timing within the bus interface logic.

2.12 DMA BOARD SCHEMATIC

Figure 9 provides a complete schematic drawing (3 sheets) of all logic on the DMA Board.

3. UTILIZATION: DMA BOARD

This section provides information on utilization of the DMA Board.

3.1 INSTALLATION

In installing the DMA Board, the user must take account of:

(a) environmental extremes
(b) mounting considerations
(c) electrical connections
(d) power requirements
(e) signal requirements
(f) base address selection
(g) interrupt level selection

Environment

Temperature extremes can cause instability, or result in permanent damage to the circuits on the

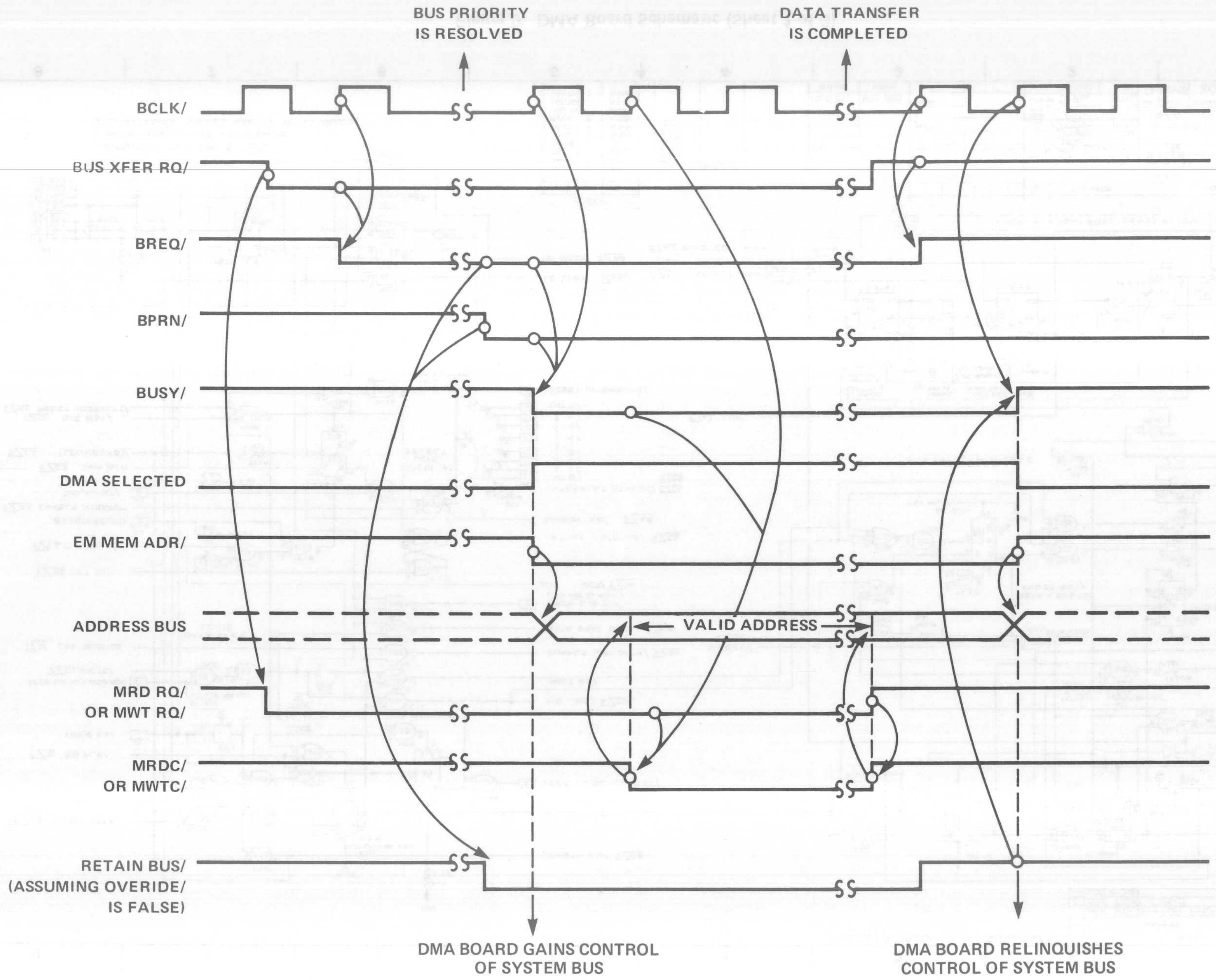
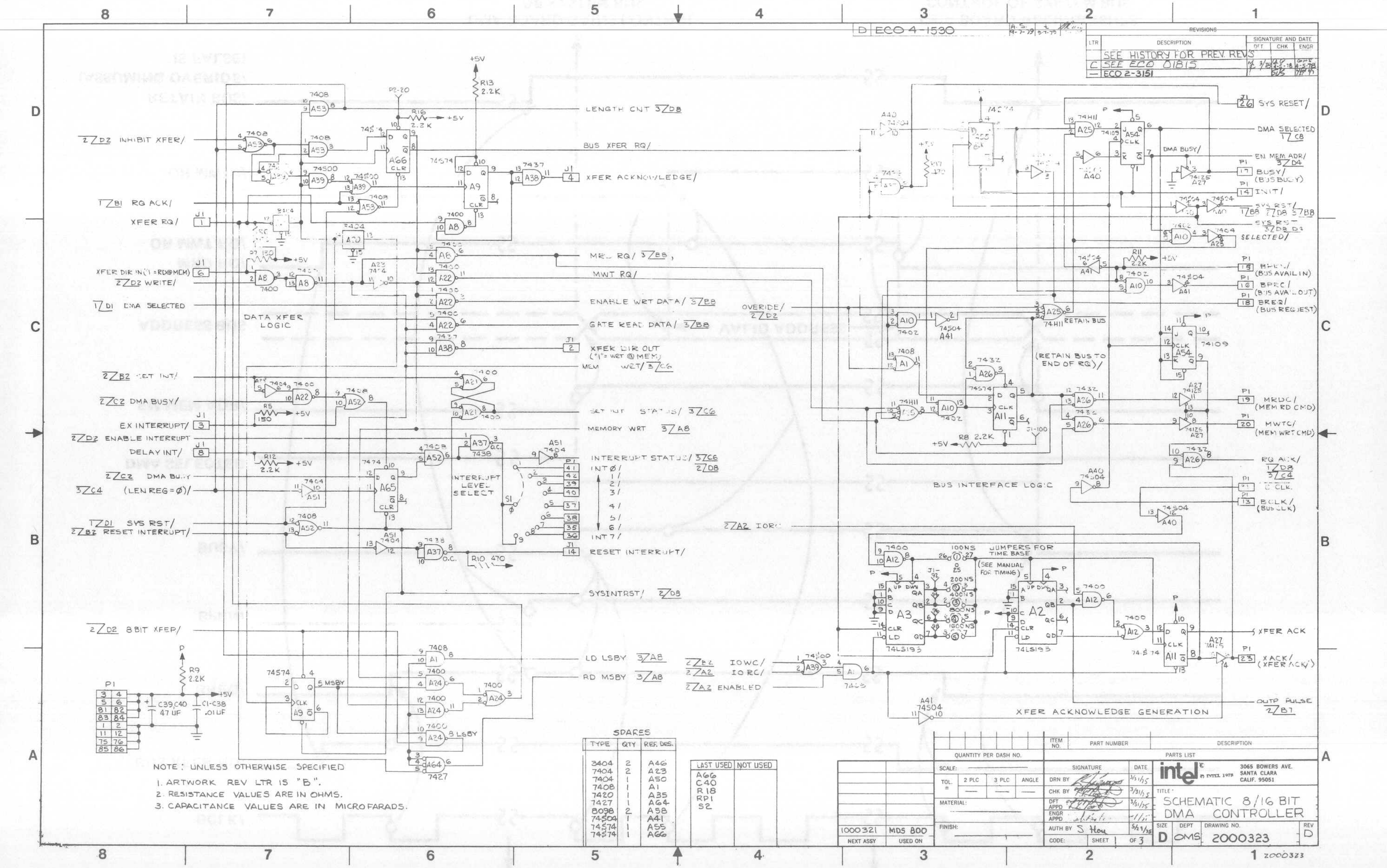

BUS PRIORITY IS RESOLVED
DATA TRANSFER IS COMPLETED
BCLK/
BUS XFER RQ/
BREQ/
BPRN/
BUSY/
DMA SELECTED
EM MEM ADR/
ADDRESS BUS
VALID ADDRESS
MRD RQ/ OR MWT RQ/
MRDC/ OR MWTC/
RETAIN BUS/ (ASSUMING OVERIDE/ IS FALSE)
DMA BOARD GAINS CONTROL OF SYSTEM BUS
DMA BOARD RELINQUISHES CONTROL OF SYSTEM BUS

Figure 8. Bus Interface Timing



SPARES

TYPE	QTY	REF. DES.
3404	2	A46
7404	2	A23
7404	1	A50
7408	1	A1
7420	1	A35
7427	1	A64
8098	2	A58
74S04	1	A41
74S74	1	A55
74S74	1	A66

LAST USED	NOT USED
A66 C40 R18 RP1 S2	

NOTE: UNLESS OTHERWISE SPECIFIED
1. ARTWORK REV LTR IS "B".
2. RESISTANCE VALUES ARE IN OHMS.
3. CAPACITANCE VALUES ARE IN MICROFARADS.

1000321	MDS 800
NEXT ASSY	USED ON

SCALE:
TOL. ± 2 PLC — 3 PLC — ANGLE —
MATERIAL:
FINISH:

SIGNATURE	DATE
DRN BY	3/31/75
CHK BY	3/31/75
DFT APPD	3/31/75
ENGR APPD	-/1/75
AUTH BY S. Hou	5/21/75
CODE: SHEET 1	OF 3

intel © INTEL 1975
3065 BOWERS AVE.
SANTA CLARA
CALIF. 95051

TITLE: SCHEMATIC 8/16 BIT DMA CONTROLLER

SIZE D DEPT OMS DRAWING NO. 2000323 REV D

Figure 9. DMA Board Schematic (Sheet 1 of 3)

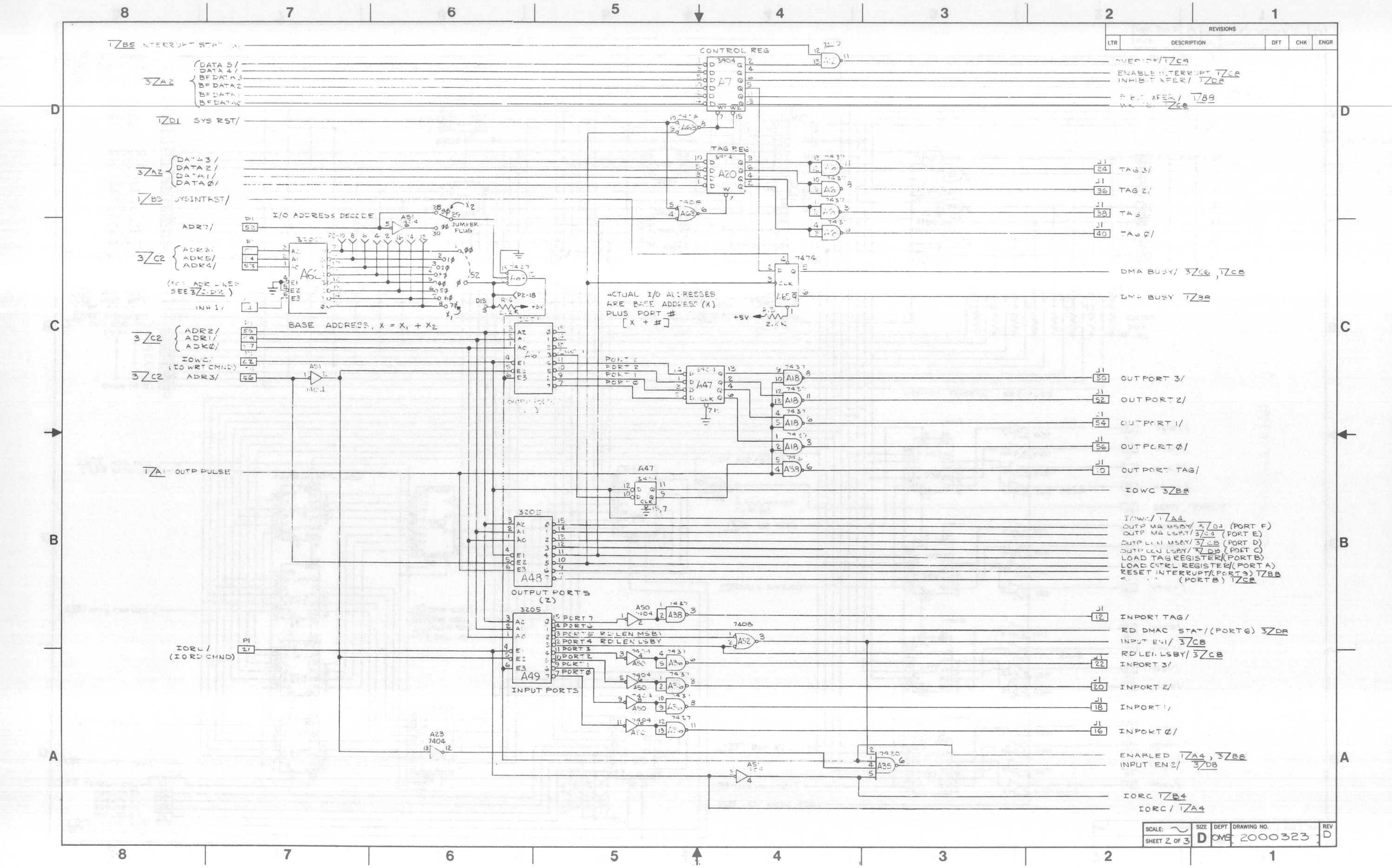


Figure 9. DMA Board Schematic (Sheet 2 of 3)

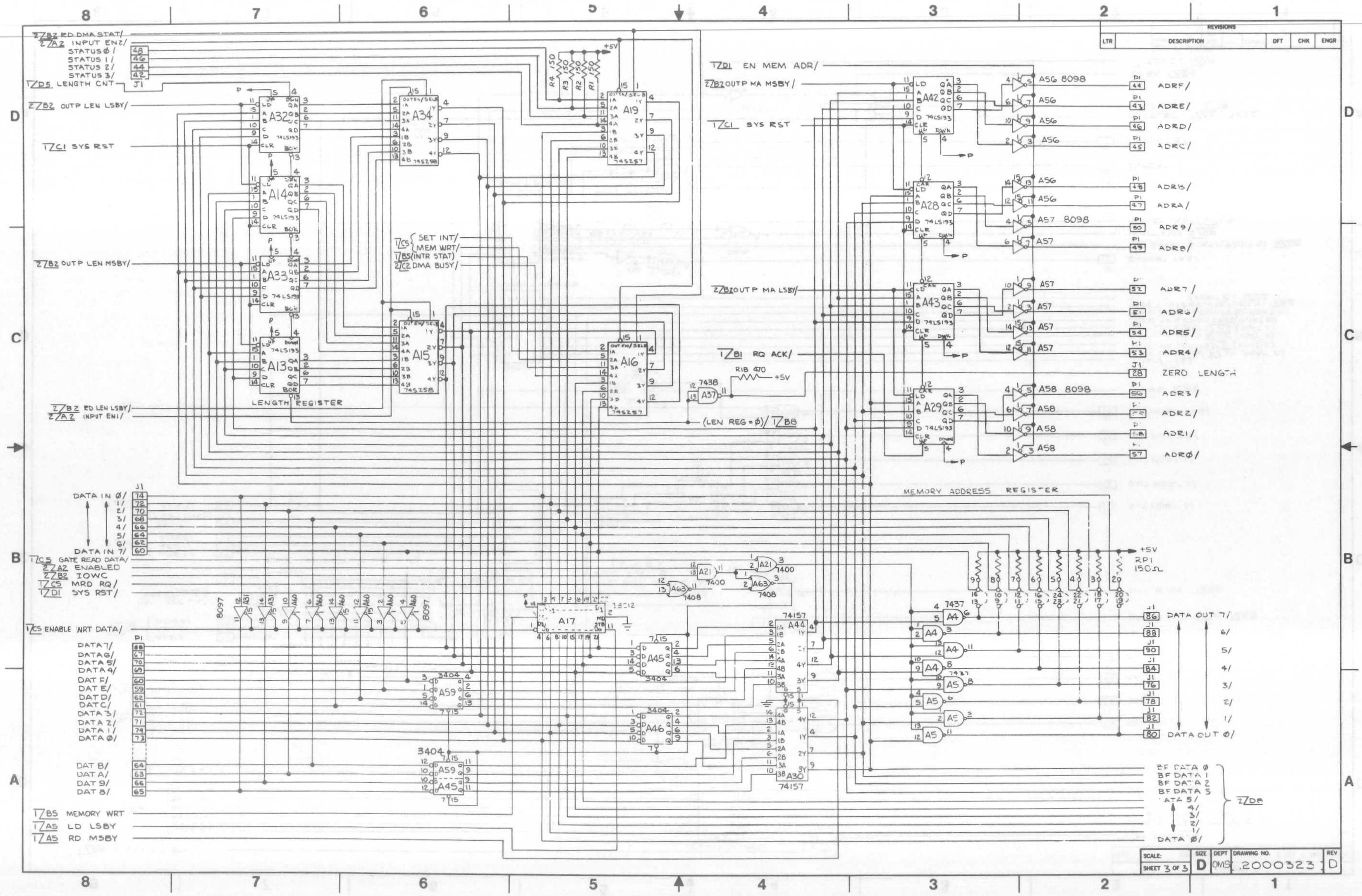


Figure 9. DMA Board Schematic (Sheet 3 of 3)

module. Ambient temperatures must therefore be maintained within the limits of 0° to 70° Centigrade. Exercise caution in locating the module, giving particular attention to radiant and conductive sources of heat. Remember that the board itself, when installed, will contribute some heat to the environment. Maintain an adequate clearance, to permit the convective dissipation of heat from the elements on the card.

Relative humidity should not exceed 90%, non-condensing.

Mounting

Avoid locating the board near vibrating machinery. Exposure to prolonged or violent vibration may cause fatigue or impact failure of connections on the board, resulting in abnormally high noise levels or outright failure of the assembly.

Dimensions of the board are 12-in. X 6.75-in. Be sure to allow enough additional clearance to ensure adequate cooling.

The board is designed to plug directly into three standard, double-sided PC edge connectors. An 86-pin connector and a 60-pin auxiliary connector are located on one edge of the board; a 100-pin connector is on the opposite edge. The connectors can serve as a mounting, as well as an electrical junction, if the environment is not too severe. Card guide slots are desirable, for the additional protection they afford. Should vibration be a problem, however, or should the assembly be used in a portable equipment application, an additional retaining bracket will have to be provided. When mounting the board, remember that it is desirable to orient the assembly vertically whenever possible. This optimizes convective cooling of the components on the board.

Electrical Connections

The DMA Board communicates with the motherboard and, consequently, the rest of the system, through a standard 86-pin, double-sided PC edge connector (P1), 0.156-in. contact centers, as shown in Figure 10. Control Data VPB01E43A00A1 is one suitable type of connector. Pin allocations on this connector are given in Table 4 of Section 3.2. The board can also communicate with other boards in the system, through the auxiliary 60-pin, double-sided PC edge connector IP2), 0.1-in. contract centers (see Figure · 10). Pin allocations for this connector (primarily test points) are listed in Table 5. The board transfers information to/from the peripheral devices via a 100-pin, double-sided PC edge connector (J1) which attaches to the edge opposite that of the other two connectors. This connector has 0.1-in. contact centers. Viking 3VH50/1JN5 is one suitable type of connector for communicating with the peripheral devices. Pin allocations for this connector are given in Table 6.

The DMA Board requires +5 VDC power.

Refer to the pin list in Table 4 of Section 3.2 for power connections.

Signal Requirements

All data and control functions appearing at the board edge connectors are at TTL levels. Electrical characteristics of the signal inputs and outputs, as well as power inputs, are given in Section 4.

Signal descriptions and connector pin allocations are given in Section 3.2.

Base Address Selection

The user must assign a base address to the DMA Board. The base address is defined by the four most significant bits of the 8-bit I/O port address. The four least significant bits, then, define 16 unique addresses within the range defined by each base address. When the CPU accesses the DMA board by executing an I/O instruction, the base address (BASE) selects the proper DMA Board, while the four low-order address bits select one of the internal DMA registers or I/O ports, as described in Section 2.1. For example, an output instruction to port "BASE+B_{16}" loads the tag register on the DMA Board, while an input instruction to address "BASE+3" reads the data byte at input port 3.

A base address is assigned by connecting the proper poles of the X2 jumper plug (28-29-30) and by adding a jumper wire between two of the wire wrap posts on X1 (S2), shown on sheet 2 of the board

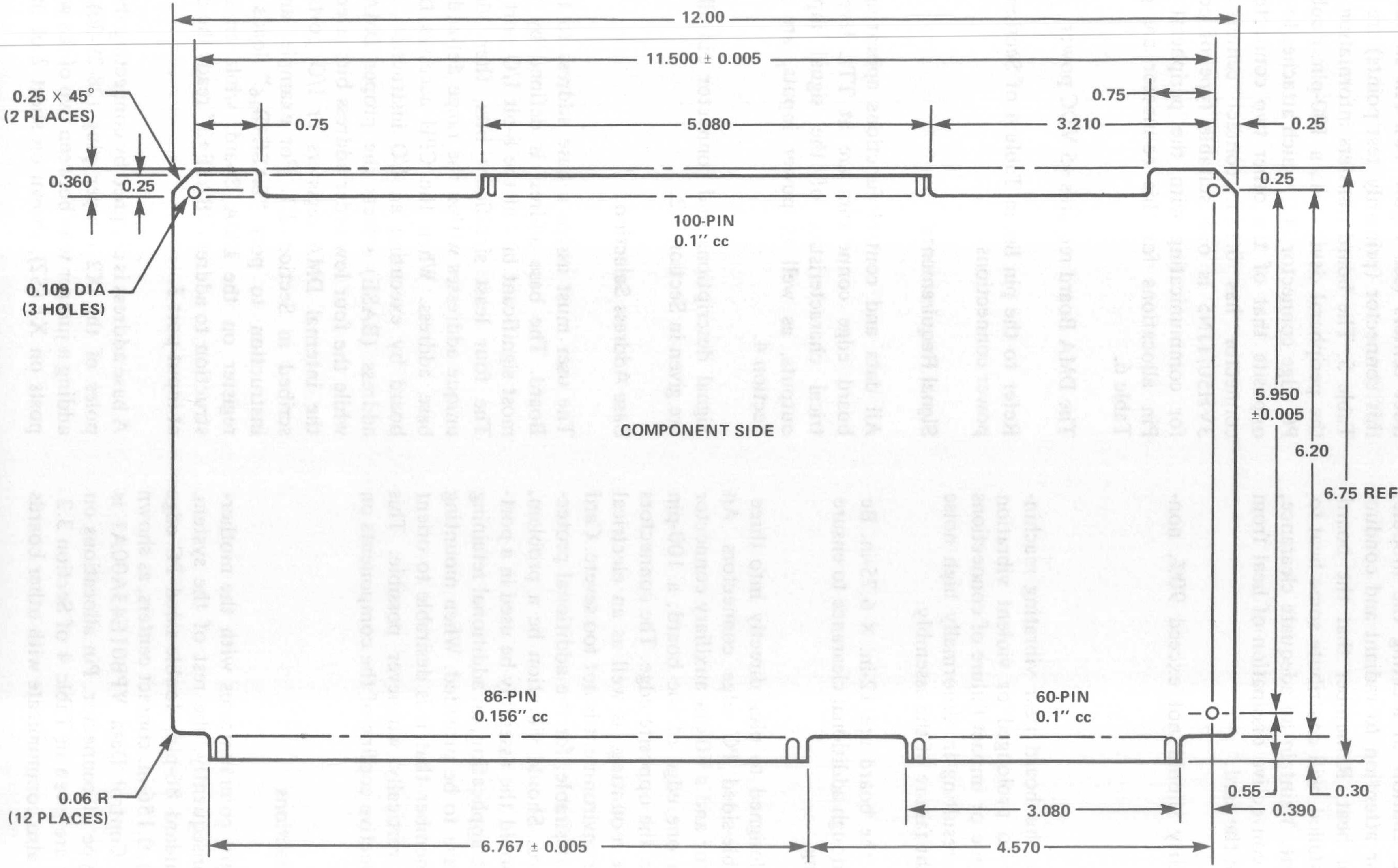


Figure 10. DMA Board Connectors

schematic. Table 1 in Section 2.1 lists the various base addresses that can be selected by connecting the X2 jumper plug and adding a jumper to X1.

Interrupt Level Selection

The user can assign the DMA Board's interrupt request line to any one of eight interrupt priority levels. (INTØ/—INT7/) by placing the interrupt level select jumper (S1) on the desired posts. The nine positions are associated with the following priority levels:

SWITCH POSITION	INTERRUPT PRIORITY LINE	RELATIVE PRIORITY (INTELLEC® SYSTEM)
S1-0 to S1-1	INTØ/	Highest
S1-0 to S1-2	INT1/	
S1-0 to S1-3	INT2/	
S1-0 to S1-4	INT3/	
S1-0 to S1-5	INT4/	
S1-0 to S1-6	INT5/	
S1-0 to S1-7	INT6/	
S1-0 to S1-8	INT7/	Lowest
S1-0 to S1-9	OFF	(No Interrupt Request)

3.2 PIN LISTS: DMA BOARD

The following section provides connector pin allocations on the DMA Board. The pins and their designated signal functions for the 86-pin connector (P1) are listed in Table 4. The same information for the 60-pin auxiliary connector (P2) is listed in Table 5. Pin and signal information for the 100-pin peripheral connector (J1) is given in Table 6.

3.3 SBC 80/10 – DMA INTERFACE

The SBC 80/10 can easily interface with one SBC bus master; the DMA Controller is such a master. This section describes the SBC 80/10's bus exchange mechanism and gives instructions on how to connect the SBC 80/10 and DMA bus exchange lines. The SBC 80/10 does not exchange the bus as describes in Section 2.11 (bus interface logic).

A minimum DMA system consists of an SBC 80/10, a DMA Controller (SBC 501), and a memory expansion board (i.e., SBC 016 or SBC 104). Since the memory on the SBC 80/10 may only be accessed by the SBC 80/10 CPU, the memory expansion board is required to provide a common link for data transfers between the CPU and the DMA device. In this two master system, the DMA Controller must have the higher bus priority. This is done by asserting BPRN/ (Bus PRiority IN/, Active Low, $V_{IN} < 0.2V$). Each time the DMA Controller needs the bus, it activates BREQ/ (Bus REQUEST/, Active Low, $V_{IN} < 0.2V$). This signal then deactivates the SBC 80/10's BPRN (Bus PRiority IN, Active High, $V_{IN} > 2.0V$) line. When BPRN goes low (taking away bus priority), the SBC 80/10 goes into a HOLD state. Once the SBC 80/10 is in the HOLD state, it releases BUSY/ (Active Low, $V_{OUT} > 2.0V$). The combination of BPRN/ low and BUSY/ high enables the DMA Controller to acquire the bus on the next bus clock cycle. When the DMA Controller completes the bus transfer, its BREQ/ signal goes high (releasing the bus), thus causing the SBC 80/10's BPRN signal also to go high. The SBC 80/10 then leaves the HOLD state and reacquires the bus. The SBC 80/10 does not use BUSY/ or bus clock to get back on the system bus. It takes about 1.1 μsec for the SBC 80/10 to drive BUSY/ true (Active Low $V_{OUT} < 0.2V$), once BPRN goes true. Thus, the DMA Controller must wait at least 1.2 μsec before requesting another transfer once it has completed a transfer.

The DMA Board can *not* be used in the override mode of operation with the SBC 80/10. The SBC 80/10 does not reacquire the bus using the BUSY signal which the DMA Board uses to hold the bus in override mode.

The following are the steps needed to interface the SBC 80/10 and the DMA Controller:

(1) Ground pin 15 (BPRN) of the connector on the motherboard that P1 of the DMA Controller is to go into.

(2) Connect pin 18 (BREQ/) of the connector on the motherboard that P1 of the DMA Controller is to go into to pin 15 (BPRN) of the connector on the motherboard that the SBC 80/10 is to go into.

(3) If the interrupt mode on the DMA Controller is to be used, the interrupt level switch on the DMA Controller must be set to INT1/ (pin 42), the bus interrupt line for the SBC 80/10.

Figure 11 shows a block diagram of the SBC 80/10 DMA system.

4. OPERATING CHARACTERISTICS: DMA BOARD

The AC and DC characteristics of all major signals that appear at the edge connectors is listed in this section.

4.1 AC CHARACTERISTICS

AC characteristics are given in Tables 7a, 7b and 8, and Figures 12 through 17.

4.2 DC CHARACTERISTICS

DC characteristics are given in Table 9.

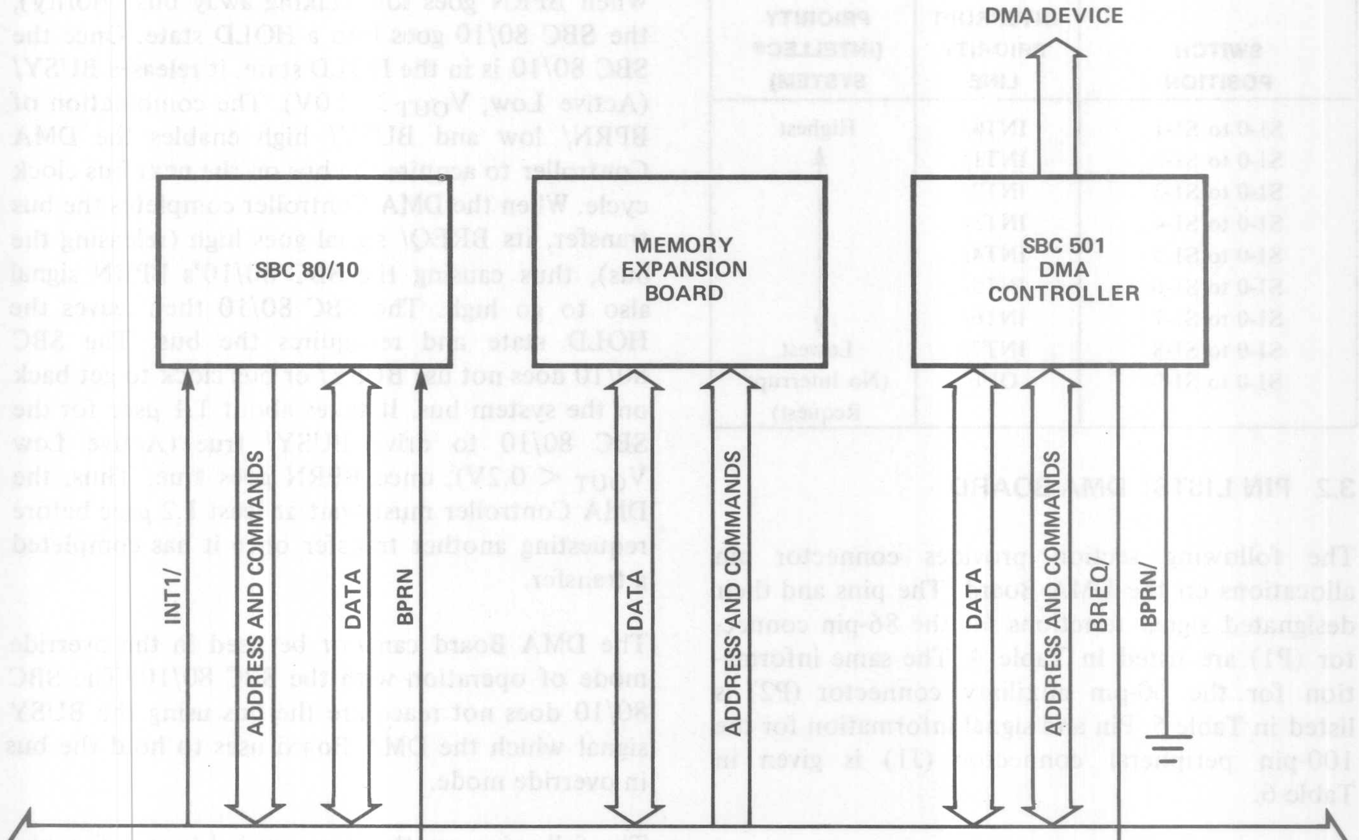


Figure 11. SBC Bus Interconnection for SBC 80/10 and SBC 501 DMA Interface

Table 4

P1 CONNECTOR PIN LIST

PIN	SIGNAL	FUNCTION	PIN	SIGNAL	FUNCTION
1	GND	Ground	44	ADRF/	Address bus
2	GND		45	ADRC/	
3	+5 VDC	Power inputs	46	ADRD/	
4	+5 VDC		47	ADRA/	
5	+5 VDC		48	ADRB/	
6	+5 VDC		49	ADR8/	
7			50	ADR9/	
8			51	ADR6/	
9			52	ADR7/	
10			53	ADR4/	
11	GND	Ground	54	ADR5/	
12	GND		55	ADR2/	
13	BCLK/	Bus clock (9.8304 MHz)	56	ADR3/	
14	INIT/	System reset	57	ADRØ/	
15	BPRN/	Bus priority in	58	ADR1/	
16	BPRO/	Bus priority out	59	DATE/	Data bus
17	BUSY/	Bus busy	60	DATF/	
18	BREQ/	Bus request	61	DATC/	
19	MRDC/	Memory read command	62	DATD/	
20	MWTC/	Memory write command	63	DATA/	
21	IORC/	I/O read command	64	DATB/	
22	IOWC/	I/O write command	65	DAT8/	
23	XACK/	Acknowledge transfer	66	DAT9/	
24			67	DAT6/	
25			68	DAT7/	
26			69	DAT4/	
27			70	DAT5/	
28	ZERO LENGTH	Indicates Len REg = ØØØØ	71	DAT2/	
29			72	DAT3/	
30			73	DATØ/	
31	CCLK/	Common clock (9.8304 MHz)	74	DAT1/	
32			75	GND	Ground
33			76	GND	
34			77		
35	INT6/	Interrupt requests	78		
36	INT7/		79		
37	INT4/		80		
38	INT5/		81	+5 VDC	Power inputs
39	INT2/		82	+5 VDC	
40	INT3/		83	+5 VDC	
41	INTØ/		84	+5 VDC	
42	INT1/		85	GND	Ground
43	ADRE/	Address bus	86	GND	

Table 5

P2 CONNECTOR PIN LIST

PIN	SIGNAL	FUNCTION
1		TEST POINTS
2	40 ADR/	
3		
4	30 ADR/	
5		
6	20 ADR/	
7		
8	10 ADR/	
9		
10	00 ADR/	
11		
12	70 ADR/	
13		
14	60 ADR/	
15		
16	50 ADR/	
17		
18	DIS ADR	
19		
20	SET BUS XFER RQ	
21		
22		
23		
24		
25		
26		
27		
28		
29		
30		
31		
32		
33		
34		
35		
36		
37		
38		
39		
40		
41		
42		
43		
44		
45		
46		
47		
48		
49		
50		
51		
52		
53		
54		
55		
56		
57		
58		
59		
60		

Table 6

J1 CONNECTOR PIN LIST

PIN	SIGNAL	FUNCTION
1	XFER RQ/	Transfer request
2	XFER DIR OUT	Transfer direction out
3	EX INTERRUPT/	External interrupt
4	XFER ACKNOWLEDGE/	Transfer acknowledge
5		
6	XFER DIR IN	Transfer direction in
7		
8	DELAY INT/	Delay interrupt
9		
10	OUTPORT TAG/	Output strobe
11		
12	INPORT TAG/	Input strobe
13		
14	RESET INTERRUPT/	Reset interrupt
15		
16	INPORTØ/	Input strobes
17		
18	INPORT1/	
19		
20	INPORT2/	
21		
22	INPORT3/	
23		
24	TAG3/	Tag register, bit 3
25		
26	SYS RESET/	System reset
27		
28		
29		
30		
31		
32		
33		
34		
35		
36	TAG2/	Tag register, bit 2
27		
38	TAG1/	Tag register, bit 1
39		
40	TAGØ/	Tag register, bit 0
41		
42	STATUS 3/	External status, bit 3
43		
44	STATUS 2/	External status, bit 2
45		
46	STATUS 1/	External status, bit 1
47		
48	STATUS 0/	External status, bit 0
49		
50	OUTPORT3/	Output strobes

Table 6

J1 CONNECTOR PIN LIST (continued)

PIN	SIGNAL	FUNCTION
51		
52	OUTPORT2/	Output strobes
53		
54	OUTPORT1/	
55		
56	OUTPORTØ/=	
57		
58		
59		
60	DATA IN7/	Data input bus (from device)
61		
62	DATA IN6/	
63		
64	DATA IN5/	
65		
66	DATA IN4/	
67		
68	DATA IN3/	
69		
70	DATA IN2/	
71		
72	DATA IN1/	
73		
74	DATA INØ/	
75		
76	DATA OUT3/	Data output bus (to device)
77		
78	DATA OUT2/	
79		
80	DATA OUTØ/	
81		
82	DATA OUT1/	
83		
84	DATA OUT4/	
85		
86	DATA OUT7/	
87		
88	DATA OUT6/	
89		
90	DATA OUT5/	
91		
92	200 ns	Test point
93		
94	400 ns	
95		
96	800 ns	
97		TEST POINTS
98	1600 ns	
99		
100	ASSERT RETAIN BUS/	

Table 7a

DMA CONTROLLER BUS MASTER AC CHARACTERISTICS

PARAMETER	OVERALL		CONTINUOUS BUS CONTROL (OVERRIDE)		EXCHANGE OF BUS CONTROL		DESCRIPTION OUTPUT LIMITS	REMARKS
	MIN (ns)	MAX (ns)	MIN (ns)	MAX (ns)	MIN (ns)	MAX (ns)		
t_{AS}	50		50		$t_{CY}-43$		Address setup time to command	$^{3}t_{CY} = t_{BCY}$ (Min Per Apl)
t_{AH}	$t_{XKO}+9$		$t_{XKO}+9$		$t_{XKO}+9$		Address hold time from command	See t_{XKO} below
t_{DS}	$^{3}t_{BCY}-59$		$^{1}50$		$^{3}t_{BCY}-59$		Data setup time to command, write	$^{1}t_{EDSW}-11_{ns}$ (Table 8-7b)
t_{DHW}	$^{2}50$		$^{2}50$		Fdu		Data hold time from command, write	$^{2}t_{EDHW}-73.5_{ns}$ (Table 8-7b)
t_{DD}							Data delay during memory write	Data always valid during command (t_{DS})
t_{SEP}	260		260				Command separation	
t_{WC}	$t_{ACC}+15$		$t_{ACC}+15$		$t_{ACC}+15$		Command width	Assuming XACK/ t_{ACC}=delay
t_{DBS}	9	33			9	33	Bus sample to exchange initiation	Sample point = BCLK/↓
t_{BS}	−29	$76+t_{BCY}$			−29	$76+t_{BCY}$	Bus sampling point delay	
t_{DB}		59				59	Data and address turn on delay	
t_{DRQ}		13.5				13.5	Bus request delay	Fig. 7b
t_{DBY}		57.5				57.5	Bus busy turn on delay	Fig. 7b
t_{DBYF}		32.5				32.5	Bus busy turn off delay	Fig. 7b
t_{DBPO}		26.5				26.5	BPRO/ serial delay from BPRN/	Fig. 7b
t_{XKCO}	15	104.5	15	104.5	15	104.5	Command turn off delay from XACK/	
							Input Requirements	
t_{XKD}	1		1		1		XACK delay from valid read data	
t_{XKO}		70		70		70	XACK turn off delay	
t_{BCY}	100		100		100		Bus clock cycle time	Fig. 7b
t_{BW}	25				25		Bus clock low and high periods	Fig. 7b
t_{DBPN}		36.5				36.5	Priority input setup time	Fig. 7b
t_{DHR}	23		23		23		Data hold from read command	
t_{ACC}		System Timeout		System Timeout		System Timeout	XACK delay from command	
t_{CCY}	100						Comm. clock cycle time	
t_{CW}	25						Com. clock low and high periods	

Table 7b

DMA CONTROLLER EXTERNAL AC CHARACTERISTICS

PARAMETER	OVERALL LIMITS$_{ns}$		CONTINUOUS BUS CONTROL (OVERRIDE)		EXCHANGE OF BUS CONTROL		DESCRIPTION OUTPUT LIMITS	REMARKS
	MIN	MAX	MIN	MAX	MIN	MAX		
t_{EXKO}		40.5		40.5		40.5	XFER ACKN/ to XFER RQ/ OFF	
t_{EACC}		159.5 + t_{ACC}		159.5 + t_{ACC}		122 + t_1	Memory Access – XFER RQ to XFER ACKN/	t_1=*t_{BUSMAX} +3t_{BCY}+ t_{ACC}
t_{ECY}	t_{EACC}+ t_{ERQSEP}						XFER Cycle	
t_{EXKO}		40.5		40.5		40.5	XFER ACKN/ OFF from XFER RQ/ RD data	
t_{EXKD}	–39		–39		–39		XFER ACKN/ delay from valid RD data	(=39ns + t_{XKD} (slave))
t_{EDHR}	10		10		10		Valid RD data after XFER RQ/ OFF	
t_{EXINTD}		93					EXT INT to INTx/ delay	
t_{EDI}		46					Delay INT/ to INTx/	
$t_{EXDI/O}$		59					XFERDIRIN to XFERDIROUT delay	
t_{DOTAG}	nt_{CCY}	190.5+ $2nt_{CCY}$					IOWC/ to valid TAGx/, OUTPORTx, XFER DIROUT	N=1,2,4,8,16 jumper selectable
t_{EDSO}	nt_{CCY} –133						Output Data Set-up to OUTPORTx/ STROBE	
t_{ESTB}	13 + $2nt_{CCY}$	122 + $2nt_{CCY}$					OUTPORTx/ STROBE width	
t_{EDHO}	nt_{CCY}						Output Data Hold from OUTPORTx/ STROBE	
t_{DII}	8	55					IORC/ to INPORTx/ delay	
t_{INPORT}	$2nt_{CCY}$ –55						INPORTx/ width	
t_{EDSW}	58		58		0		INDATA and XFERDIRIN set-up to XFER RQ/	
t_{ERQ}	t_{EACC}		t_{EACC}				XFER RQ/ width	t_{ERQ} Min = t_{EACC} t_{EXKCO} Min
t_{ERQSEP}	181.5*		**		181.5		XFER RQ/ Separation	
t_{EDHW}	123.5		123.5				INDATA XFERDIRIN hold time after XFER RQ/	
t_{EXKCO}	0		0		0		XFER ACKN/ to XFER RQ/ OFF	
t_{EACCI}		$2n_{TC}$ –115					DATAINx/ valid from INPORTx/	n=1,2,4,8,16 jumper selectable
		85						t_{CCY}=100ns,n=1
		285						t_{CCY}=100ns,n=2
		685						t_{CCY}=100ns,n=4
		1485						t_{CCY}=100ns,n=8
		3085						t_{CCY}=100ns,n=16

*In SBC 80/10 System, $t_{ERQSEP\ MIN}$ = 1.2 μsec.

**Override mode can not be used in SBC 80/10 System.

Table 7b

DMA CONTROLLER EXTERNAL AC CHARACTERISTICS (continued)

PARAMETER	OVERALL LIMITS$_{ns}$		CONTINUOUS BUS CONTROL (OVERRIDE)		EXCHANGE OF BUS CONTROL		DESCRIPTION OUTPUT LIMITS	REMARKS
	MIN	MAX	MIN	MAX	MIN	MAX		
t_{EDHI} -	0		0		0		DATAINx/ hold from INPORTy/	
t_{EXTINT}	46						EXTINT/ width	*t_{BUSMAX}= longest time a BUS master will keep BUSY/ true.

Table 8

DMA CONTROLLER BUS SLAVE AC CHARACTERISTICS

PARAMETER	OVERALL		DESCRIPTION OUTPUT LIMITS	REMARKS
	MIN	MAX		
t_{DHR}	0		REFER TO FIGURE 12	
t_{XKD}	50			
t_{XKO}	40.5			
			INPUT REQUIREMENTS	
t_{AS}	33		REFER TO FIGURE 12	
t_{AH}	0.5			
t_{DS}	0			
t_{DHW}	50			
t_{WC}	t_{ACC}			
t_{CSEP}	50			
t_{ACCW}		$5nt_{CCY}$ +78.5		n=1,2,4,8,16
t_{ACCR}		$3nt_{CCY}$ +78.5		n=1,2,4,8,16
t_{XKCO}	0			

Table 9

DMA CONTROLLER DC CHARACTERISTICS

SIGNAL	SYMBOL	PARAMETER DESCRIPTION	TEST DESCRIPTION	PARAMETER			UNITS
				MIN	TYP	MAX	
ADRØ/ADRF/ ADDRESS	V_{OL}	Output low voltage	I_{OL} = 32 mA			0.4	V
	V_{OH}	Output high voltage	I_{OH} = –5.2 mA	2.4			V
	C_L	Capacitive Load				15	pF
MRDC/,MWTC/ MEMORY COMMAND	V_{OL}	Output low voltage	I_{OL} = 16 mA			0.4	V
	V_{OH}	Output high voltage	I_{OH} = –2 mA	2.4	3.1		V
	I_{LH}	Output leakage high	HIGH Z V_0 = 2.4 V			40	µA
	I_{LL}	Output leakage low	HIGH Z V_0 = 0.4 V			–40	µA
	C_L	Capacitive load				15	pF
IORC/,IOWC/ I/O COMMANDS	V_{IL}	Input low voltage				0.8	V
	V_{IH}	Input high voltage		2			V
	I_{IL}	Input current at V_{IL}	V_{IL} = 0.5 V			–3.85	mA
	I_{IH}	Input current at V_{IH}	V_{IH} = 2.7 V			170	µA
	C_L	Capacitive load				15	pF
INTØ/-INT7/ INTERRUPTS (One line only, switch selected)	V_{OL}	Output low voltage	I_{OL} = 48 mA			0.4	V
	I_{OH}	Output leakage high	Open col output is off V_{OH} = 5.5 V			100	µA
	C_L	Capacitive load				15	pF
DATØ/-DATF/	V_{OL}	Output low voltage	I_{OL} = 15 mA			.45	V
	V_{OH}	Output high voltage	I_{OH} = –1 mA	2.4			V
	V_{IL}	Input low voltage				.85	V
	V_{IH}	Input high voltage		2.0			V
	C_L	Capacitive load				15	pF
	I_{IL}	Input current at V_{IL}	V_{IN} = 0.5 V			0.65	mA
	I_{IH}	Input current at V_{IH}	V_{IN} = 2.4 V			220	µA
INIT/	V_{IL}	Input low voltage				0.8	V
	V_{IH}	Input high voltage		2			V
	I_{IL}	Input current at V_{IL}	V_{IN} = 0.5 V			–12.8	mA
	I_{IH}	Input current at V_{IH}	V_{IN} = 2.7 V			410	µA
	C_L	Capacitive load				50	pF
XACK/,BUSY/	V_{OL}	Output low voltage	I_{OL} = 16 mA			0.4	V
	V_{OH}	Output high voltage	I_{OH} = –2 mA	2.4			V
	V_{IL}	Input low voltage				0.8	V
	V_{IH}	Input high voltage		2			V
	I_{IL}	Input current at V_{IL}	V_{IL} = 0.4 V			–2.04	mA
	I_{IH}	Input current at V_{IH}	V_{IH} = 2.4 V			90	µA
	C_L	Capacitive load				15	pF
+5 VOLTS	I_{+5V}	+5 Volts Supply Current	Worst Case Component Analysis			3.35	A

Table 9

DMA CONTROLLER DC CHARACTERISTICS (continued)

SIGNAL	SYMBOL	PARAMETER DESCRIPTION	TEST DESCRIPTION	PARAMETER			
				MIN	TYP	MAX	UNITS
ADRØ/-ADR7/	I_{IL}	Input current at V_{IL}	V_{IL} = 0.45 V			–3.2	mA
	I_{IH}	Input current at V_{IH}	V_{IH} = 2.4 V			120	μA
	V_{IL}	Input low voltage	T_A = 25°C			0.8	V
	V_{IH}	Input high voltage		2.0			V
ADR8/-ADRF/	I_{LH}	Output leakage high	HIGH Z V_0 = 2.4 V			40	μA
	I_{LL}	Output leakage low	HIGH Z V_0 = 0.4 V			–40	μA
BPRN/	V_{IL}					0.8	V[
	V_{IH}			2			V
	I_{IL}		V_{IL} = 0.4			–8.6	mA
	I_{IH}		V_{IH} = 2.4 V			170	μA
	C_L					15	pF
BPRO/,BREQ/	V_{OL}		I_{OL} = 18 mA			0.5	V
	V_{OH}		I_{OH} = –1 mA	2.7			V
	C_L					15	pF
CCLK/,BCLK/	V_{IL}					0.8	V
	V_{IH}			2			V
	I_{IL}					–2	mA
	I_{IH}					50	μA
	C_L					15	pF
INH1/	V_{IL}					0.85	V
	V_{IH}			2.0			V
	I_{IL}		V_{IL} = 0.45 V			–0.25	mA
	I_{IH}		V_{IH} = 5.25 V			10	μA
	C_L					15	pF

NOTE: Test conditions include loading by the DMA board itself for bidirectional signals.

NOTE: V_{CC} = MIN, T_A = 0° to 70°C unless otherwise stated.

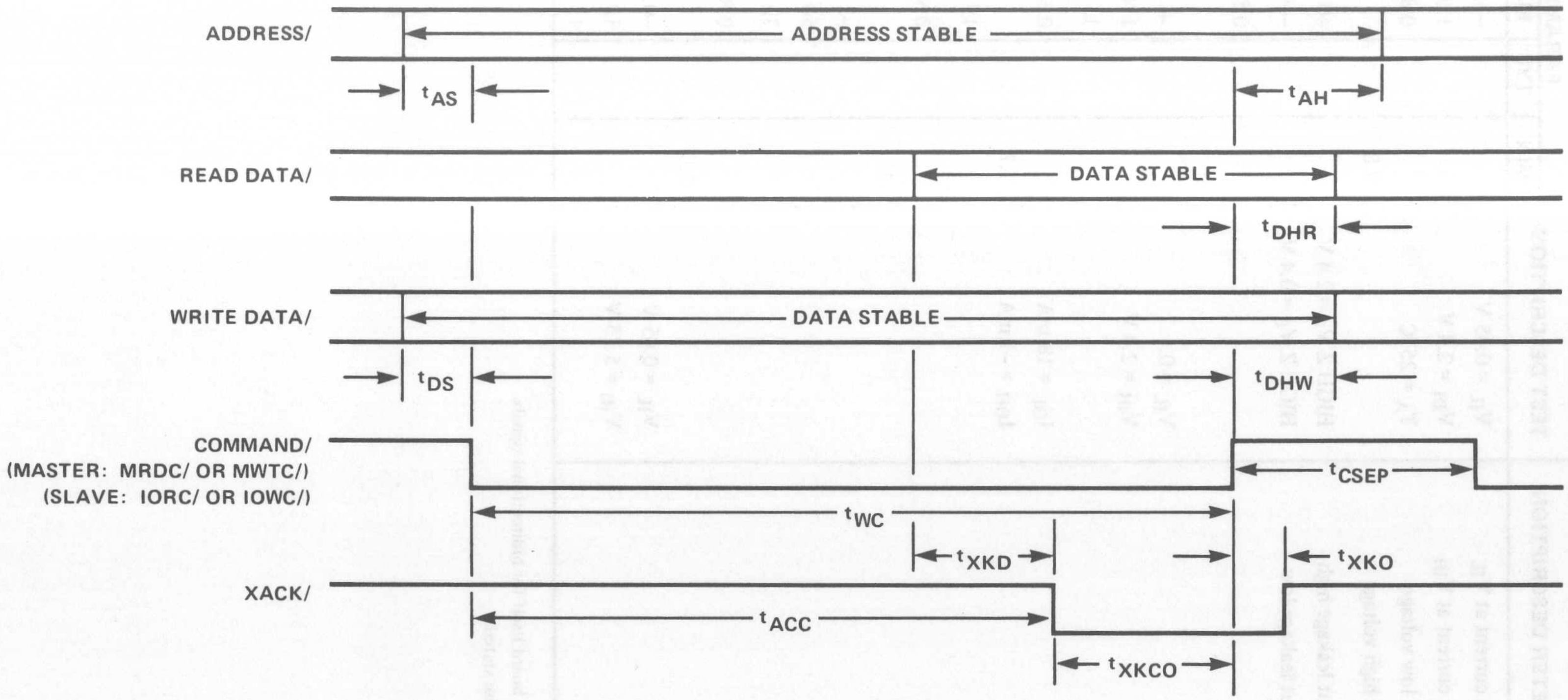


Figure 12. Command Timing – Master: Continuous Bus Control (Override Mode)

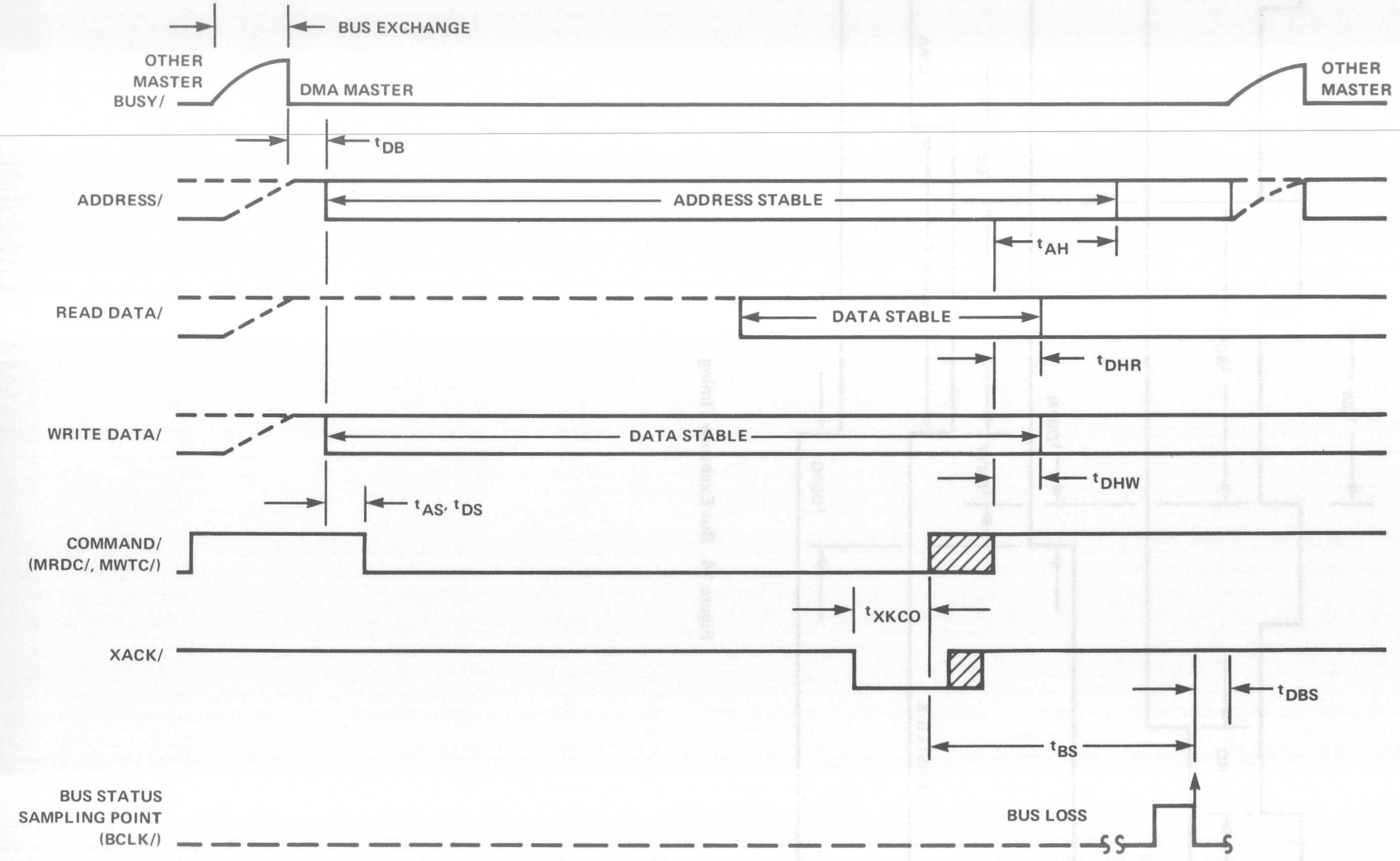


Figure 13. Master Command Timing – Bus Exchange (Shared Bus Mode)

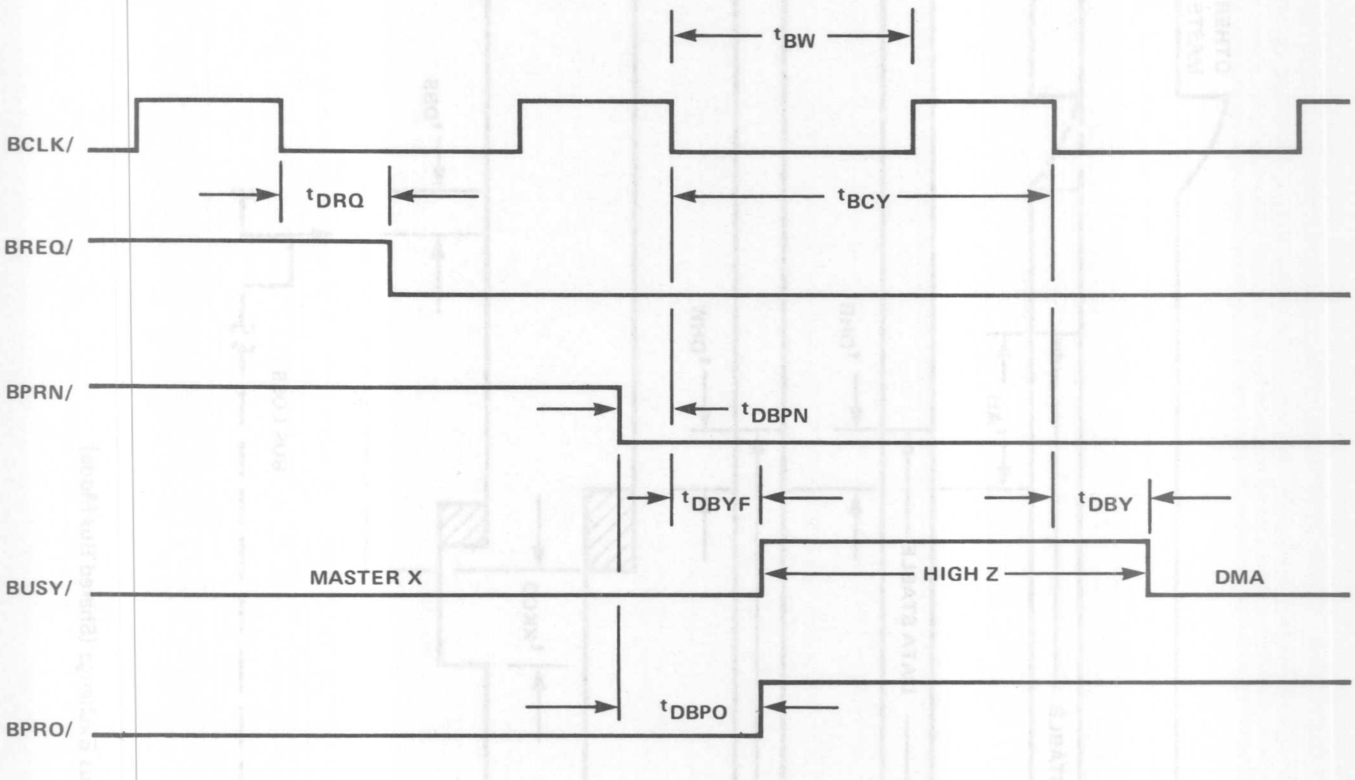


Figure 14. Bus Exchange Timing

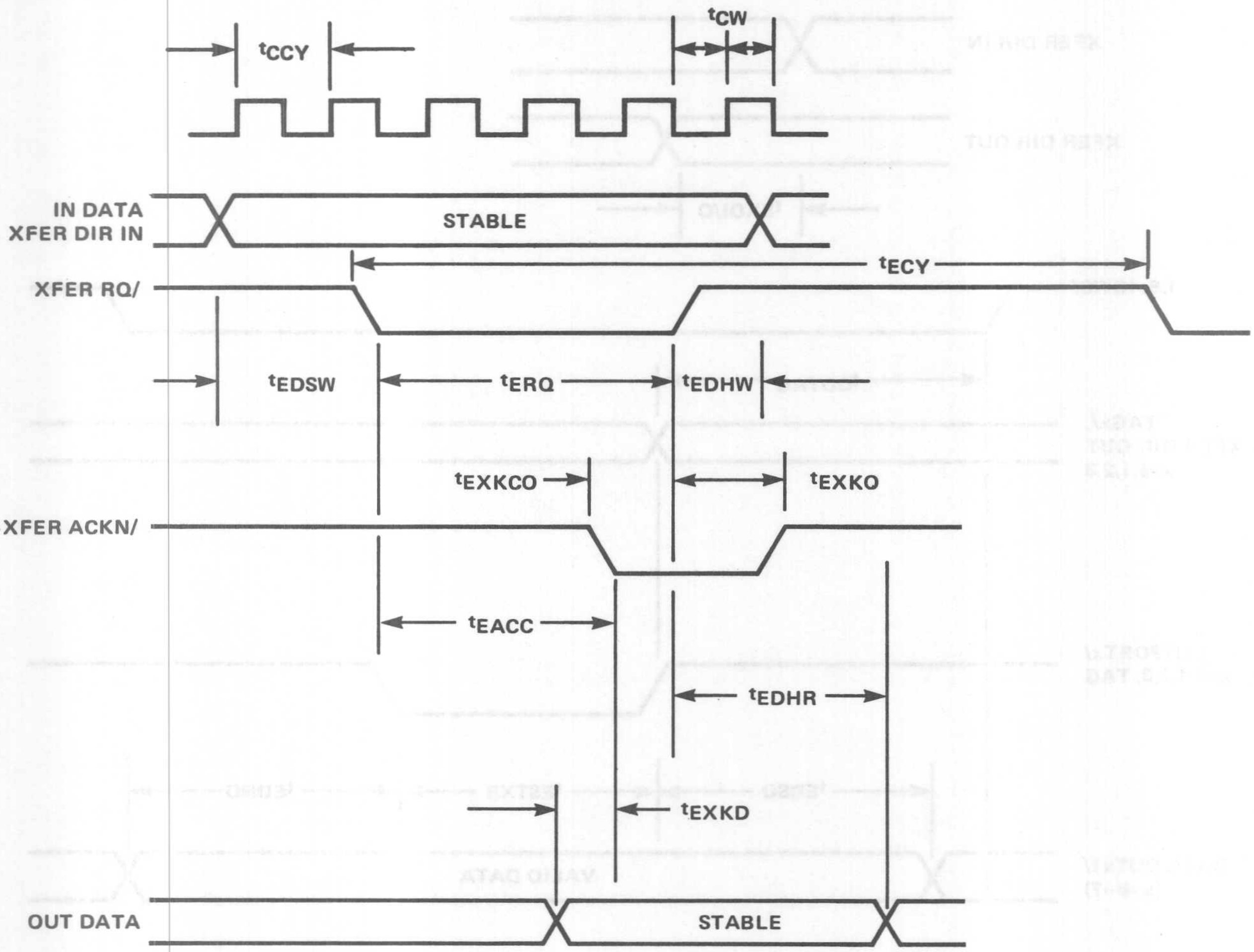


Figure 15a. External Transfer Timing

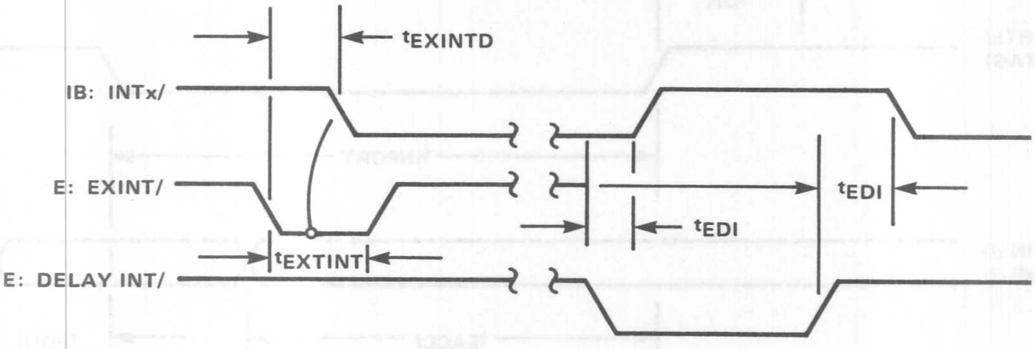


Figure 15b. Interrupt Timing

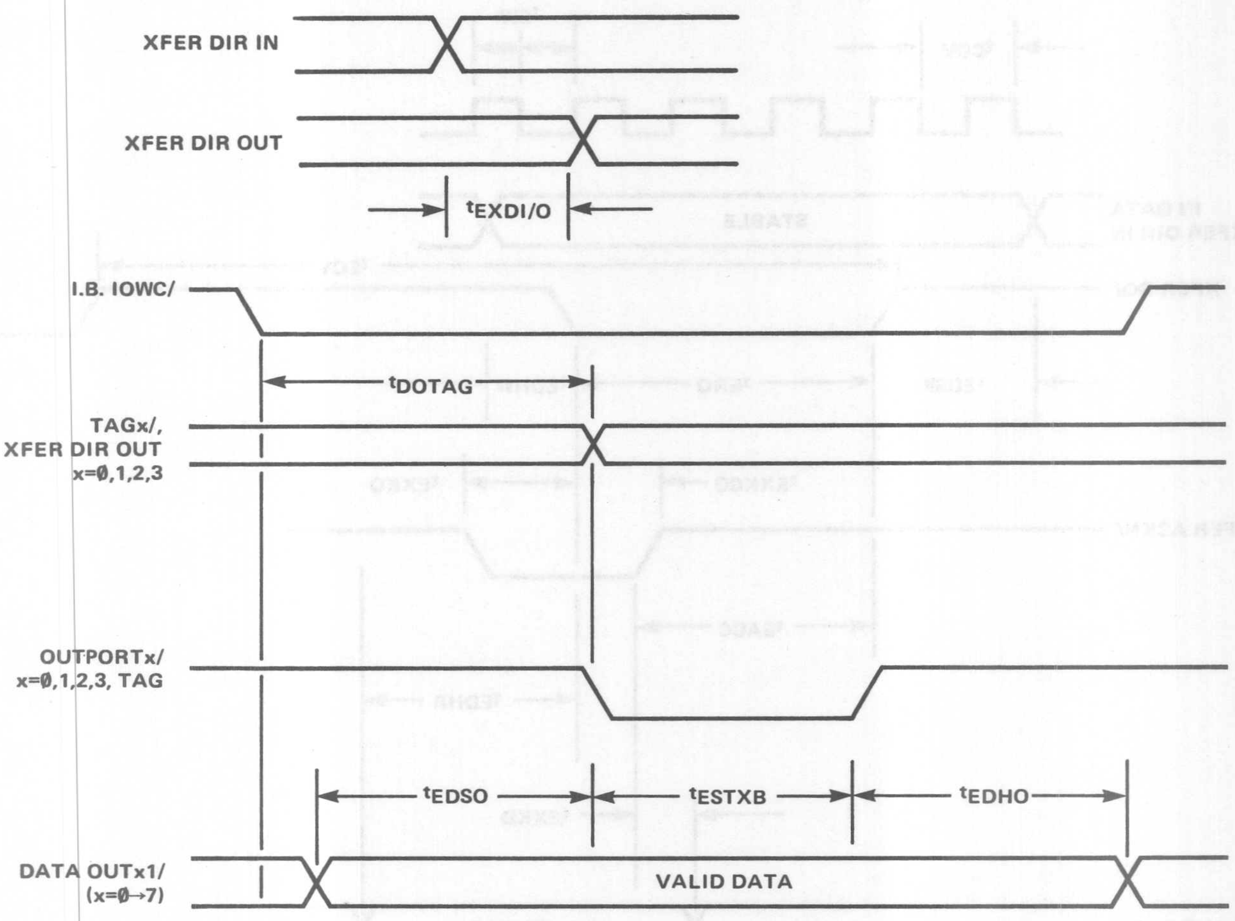


Figure 16. External I/O Write Timing

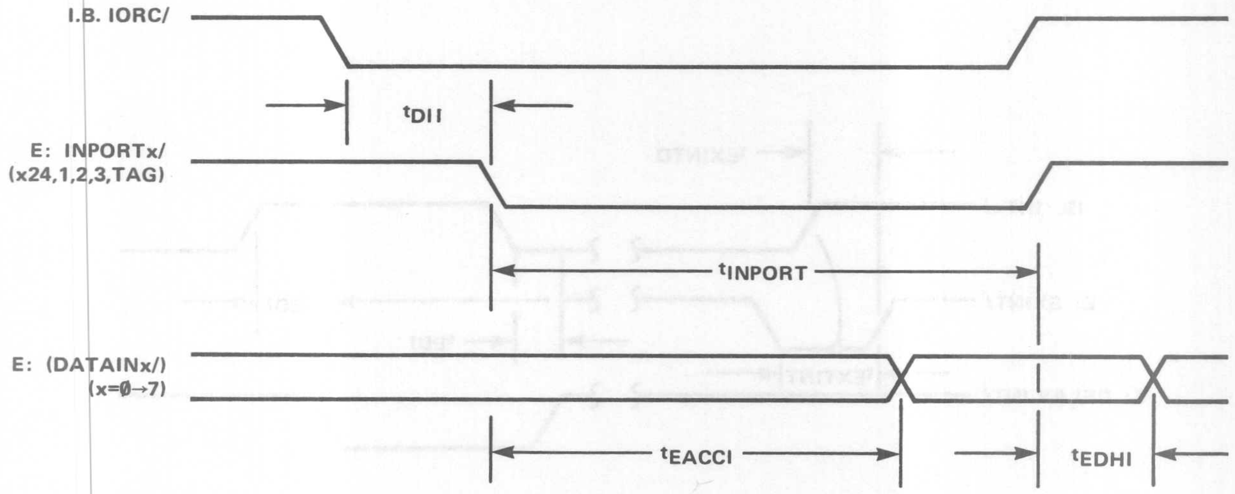


Figure 17. External I/O Read Timing

Table 10

EXTERNAL INTERFACE DMA DC CHARACTERISTICS

SIGNAL	SYMBOL	PARAMETER DESCRIPTION	TEST CONDITIONS	PARAMETER		
				MIN	MAX	UNITS
XFER RQ/, XFER DIR IN/, EX INTERRUPT/, STATUSx/ (x=0,1,2,3), DATA IN y/ (y=0→7)	V_{IL}	Input Low Voltage			0.4	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at V_{IL}	$V_{IL} = 0.5$ V		–44.0	mA
	I_{IH}	Input Current at V_{IH}	$V_{IH} = 2.7$ V		190	μA
	C_L	Capacitive Load			15	pF
DELAY INT/	V_{IL}				0.4	V
	V_{IH}			2.0		V
	I_{IL}		$V_{IL} = 0.5$ V		–4.0	mA
	I_{IH}		$V_{IH} = 2.4$ V		40	μA
	C_L				15	pF
XFER ACKNOWLEDGE/ XFER DIR OUT, TAGx (x=0,1,2,3), OUTPUTy (y=0,1,2,3,TAG), INPUT y, DATA OUT z (z=0→7)	V_{OL}	Output Low Voltage	$I_{OL} = 48$ mA		0.4	V
	V_{OH}	Output High Voltage	$I_{OH} = -1.2$ mA	2.4		V
	C_L				15	pF
RESET INTERRUPT	V_{OL}	Output Low Voltage	$I_{OL} = 36$ mA		0.4	V
	V_{OH}	Output High Voltage	$I_{OH} = -1.2$ mA	2.4		V
	C_L				15	pF